

腾讯健康数据服务

API 文档



腾讯云

【 版权声明 】

©2013–2026 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 商标声明 】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

文档目录

API 文档

简介

API 概览

调用方式

资源基础操作接口

创建

更新

删除

补丁

Read

vRead

搜索操作接口

基础搜索接口

引用搜索接口

数值搜索接口

日期时间搜索接口

分页与结果数量控制

排序搜索

患者全量信息检索

Bundle 请求接口

基础 Bundle 事务

捆绑多个关联资源

占位符 ID 与引用

条件创建

条件更新

事务内删除

事务内 Patch

错误码

API 文档

简介

最近更新时间：2026-07-28 16:18:00

概述

本文档面向接入和使用本平台 FHIR API 的开发者、集成实施人员及业务技术团队，介绍平台提供的标准化医疗健康数据存储与访问能力，并说明接口调用、资源操作、搜索检索、事务处理及错误响应等内容。

本平台提供基于标准 FHIR API 的数据服务能力，支持对医疗健康数据进行统一建模、标准化访问和结构化交换。调用方可通过标准 HTTP 方法对 FHIR 资源执行创建、读取、更新、删除、局部更新、条件搜索以及批量事务处理等操作，从而满足临床应用集成、科研数据治理、跨系统数据交换和业务服务化等场景需求。

平台接口采用资源化和 RESTful 风格设计，资源内容以标准 FHIR 结构进行组织，便于围绕 Patient、Observation、Encounter 等核心资源构建业务流程。同时，平台支持基于访问令牌的鉴权机制，确保接口调用过程中的身份认证与访问控制要求得到满足。

术语信息

下表汇总了调用本平台 FHIR API 时常见的主要概念与术语，便于在阅读后续章节前建立统一认识。

术语	说明
FHIR	Fast Healthcare Interoperability Resources，医疗健康领域的数据交换与互操作标准。本平台接口遵循该标准进行资源建模与交互。
baseUrl（服务基地址）	FHIR 服务的根访问地址，例如 <code>https://HOSTNAME/INSTANCE_ID/fhir</code> ，其中 <code>HOSTNAME</code> 为 FHIR 服务访问域名， <code>INSTANCE_ID</code> 为 FHIR 服务实例 ID。腾讯云 SaaS 场景请在 腾讯健康数据服务控制台 > 实例概览中获取，独立部署场景以部署方提供的服务访问域名和实例 ID 为准。资源读写、搜索及 Bundle 提交均基于此地址构造请求 URL。
Content-Type	请求体或响应体的 MIME 类型。写入类请求通常为 <code>application/fhir+json</code> ；Patch 请求通常为 <code>application/json-patch+json</code> 。
Resource（资源）	平台存储和访问医疗健康数据的基本单元，以标准 FHIR JSON 结构表达，例如患者、检验结果、就诊记录等。
ResourceType（资源类型）	资源的类别名称，用于区分不同业务实体，如 <code>Patient</code> 、 <code>Observation</code> 、 <code>Encounter</code> 。请求 URL 中通常以资源类型作为路径段。
id（资源 ID）	资源在平台内的唯一标识。读取、更新、删除等操作通常通过 <code>[baseUrl]/[resourceType]/[id]</code> 访问目标资源。

Reference（引用）	资源之间的关联关系，通常以 <code>ResourceType/id</code> 形式指向另一条资源，例如 <code>Patient/199963</code> 。
ETag / versionId（版本标识）	标识资源当前版本的信息。创建或更新成功后，响应头中的 <code>ETag</code> 与资源 <code>meta.versionId</code> 可用于版本控制与并发更新判断。
<code>_history</code>	资源历史版本访问路径后缀。通过 <code>[baseUrl]/[resourceType]/[id]/_history/[versionId]</code> 可读取指定历史版本。
Bundle	一种特殊的 FHIR 资源，用于封装多条记录或多次交互。搜索接口返回结果 Bundle；事务/批处理接口通过提交 Bundle 批量执行操作。
Placeholder ID（占位符 ID）	事务 Bundle 中为尚未落库资源分配的临时标识，通常采用 <code>urn:uuid:{uuid}</code> 形式，用于在同一请求内建立资源引用关系。
transaction / batch	Bundle 的两种处理模式。 – transaction：表示原子事务，全部条目要么一起成功，要么一起失败。 – batch：表示批处理，各条目独立执行。
Search Parameter（搜索参数）	附加在资源类型 URL 后的查询参数，用于按条件筛选资源，例如 <code>?name=张三&gender=male</code> 。
OperationOutcome	用于描述操作结果、校验信息或错误详情的 FHIR 资源，常见于删除失败或业务校验未通过等场景。
API Key	调用方身份标识，用于向鉴权服务申请访问令牌。可在创建 FHIR 服务实例后于 API 密钥管理页面获取。
API Secret	调用方身份密钥，与 API Key 配合使用，用于计算请求签名。请妥善保管，避免泄露。
Signature（签名）	基于 <code>API Secret</code> 对 <code>apiKey + timestamp</code> 进行 HMAC-SHA256 计算得到的签名值，用于鉴权接口的身份校验。
AccessToken（访问令牌）	调用鉴权接口成功后返回的 JWT 令牌，用于访问 FHIR 业务接口。
Bearer Token	访问 FHIR 接口时在 HTTP 请求头中携带令牌的方式，格式为 <code>Authorization: Bearer</code> 。

内容说明

本文档主要包含以下内容：

- [API 概览](#)：说明平台 API 的整体能力范围、主要接口分类、调用方式概述及典型适用场景。

- **调用方式**：说明请求结构、公共参数、接口鉴权流程以及返回结果的通用规则。
- FHIR 操作相关接口：说明 **资源基础操作**、**搜索操作** 以及 **Bundle 请求接口** 等能力。
- **错误码**：汇总常见错误码及其典型触发场景，便于调用方排查问题。

通过阅读本文档，调用方可以快速了解平台接口体系，并按照统一规范完成接口接入、资源访问、数据检索与批量处理。

整体调用流程

调用方在访问 FHIR 资源接口前，整体调用流程如下：

1. 创建 FHIR 服务实例并创建服务 API 密钥。调用方保存 `API Key` 和 `API Secret`。
2. 使用 `API Key`、`API Secret` 和时间戳生成签名。
3. 调用鉴权接口获取基于 JWT 的 `AccessToken`。
4. 使用 `AccessToken` 访问 FHIR RESTful 接口。
5. FHIR 服务向鉴权服务校验 `AccessToken`。
6. 鉴权通过后返回业务结果，否则拒绝请求。

API 概览

最近更新时间：2026-07-28 16:18:00

THDS 提供基于 FHIR R4 标准的医疗健康数据存储与访问服务，支持资源级数据写入、读取、检索以及批量事务处理能力。接口设计遵循标准化资源模型与 RESTful 风格，便于与医院信息系统、临床应用、科研平台及第三方集成系统进行对接。

能力范围

平台当前提供的 FHIR 操作接口主要覆盖以下三类能力：

- **CRUD 操作能力：**支持对标准 FHIR 资源执行创建、读取、更新、删除、局部更新以及历史版本读取等操作，适用于单条资源的生命周期管理。
- **搜索与检索能力：**支持基于条件表达式的资源搜索，包括基础搜索、引用搜索、数值搜索、日期时间搜索、分页、排序以及患者全量信息检索等场景。
- **Bundle 事务与批处理能力：**支持通过 `Bundle` 在单次请求中提交多个资源交互操作，并支持 `transaction` 与 `batch` 两种处理模式，适用于批量写入、关联资源提交、条件创建、条件更新等场景。

接口特点

- **标准化：**接口遵循 FHIR 资源模型与交互语义，便于与已有 FHIR 生态对接。
- **资源化：**所有数据通过标准资源进行表达和访问，例如 `Patient`、`Observation`、`Encounter` 等。
- **可扩展：**支持通过标准搜索参数、事务 Bundle 及资源引用机制实现更复杂的数据交互流程。
- **易集成：**基于 HTTP 与 JSON 进行交互，便于被 Web 应用、服务集成平台及业务系统调用。

主要接口分类

资源基础操作接口

CRUD 接口用于完成资源的基础生命周期管理，适用于患者主数据维护、检验结果更新、诊疗记录归档及历史版本追溯等场景。主要有以下接口：

接口	说明
Create	创建新的资源实例。
Read	读取资源的最新版本内容。
vRead	读取资源的指定历史版本内容。
Update	更新指定资源的完整内容。
Patch	对资源局部字段执行增量修改。

Delete

删除指定资源，通常为逻辑删除。

搜索操作接口

搜索接口用于按条件筛选资源数据，适用于患者检索、就诊记录查询、检验结果筛选、诊断文本匹配及全量病历聚合等场景。支持以下典型检索方式：

接口	说明
基础搜索	按单个或多个常规字段检索资源。
引用搜索	按资源之间的关联关系进行检索。
数值搜索	面向检验值、测量值等数值字段进行过滤。
日期时间搜索	面向日期、时间范围及更新时间进行过滤。
分页与结果数量控制	支持大结果集分页获取。
排序搜索	支持按指定字段排序返回结果。
患者全量信息检索	支持通过 <code>\$everything</code> 获取患者全景数据。

Bundle 请求接口

Bundle 接口用于在单次请求中提交多个操作，适用场景包括批量导入数据、一次性写入关联资源、避免重复建档及减少接口调用次数等场景。主要有以下接口：

接口	说明
基础 Bundle 事务	通过单个 <code>Bundle</code> 请求提交一个或多个 FHIR 交互操作。
捆绑多个关联资源	在同一 Bundle 中提交多条存在业务关联的资源。
占位符 ID 与引用	使用临时占位符 ID 建立同一事务内资源之间的引用关系。
条件创建	按业务条件判断目标资源是否存在，不存在时创建资源。
条件更新	按业务条件定位目标资源，并对匹配资源执行更新。
事务内删除	在事务 Bundle 中对指定资源执行删除操作。
事务内 Patch	在事务 Bundle 中对指定资源执行局部更新。

调用方式

最近更新时间：2026-07-28 16:18:01

本文档说明平台 FHIR Restful 接口的通用调用方式，包括请求结构、公共参数、接口鉴权以及返回结果。调用方在访问 FHIR 资源接口前，需先通过 API 密钥和签名机制获取 `AccessToken`，再以 `Bearer Token` 方式访问业务接口。

请求结构

平台调用过程分为两个阶段：

- 调用鉴权服务，使用 `API Key`、时间戳和签名换取 `AccessToken`。
- 在访问 FHIR Restful 接口时，通过 `Authorization: Bearer <AccessToken>` 携带访问令牌。

鉴权服务请求结构

请求方式：

```
POST /cmd/GetAccessToken HTTP/1.1
Host: <FHIR_SERVICE_HOST>
Content-Type: application/json
```

❗ 说明：

- `<FHIR_SERVICE_HOST>` 为鉴权服务访问域名，请以腾讯健康数据服务控制台实例接入信息或部署方提供的服务访问地址为准。
- 示例中的 `<FHIR_SERVICE_HOST>` 仅为占位符，调用时需替换为实际可访问域名，不要直接使用占位符发起请求。

请求体结构：

```
{
  "header": {
    "version": "v0.1",
    "flag": 0
  },
  "body": {
    "seq": 0,
    "cmd": "",
    "token": "",
    "traceid": ""
  }
}
```

```
{
  "client": {
    "platform": 0,
    "env": "",
    "isTourist": 0,
    "product": 0
  },
  "payload": {
    "timestamp": 1730000000000,
    "apiKey": "<API_KEY>",
    "signature": "<SIGNATURE>",
    "scopes": [],
    "context": {}
  }
}
```

❗ 说明:

- `timestamp` 为毫秒级 Unix 时间戳。
- `signature` 为基于 `API Secret` 计算得到的签名值。
- `scopes` 和 `context` 为扩展字段，可按实际授权场景传入。

FHIR Restful 请求结构

请求方式根据接口能力不同而变化，常见方法如下：

- `POST`：创建资源，或提交 `Bundle` 事务/批处理请求。
- `GET`：读取资源、读取历史版本、执行搜索。
- `PUT`：更新完整资源内容。
- `PATCH`：更新局部字段。
- `DELETE`：删除指定资源。

通用请求示例：

```
GET /INSTANCE_ID/fhir/Patient/199963 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

其中，`HOSTNAME` 为 FHIR 服务访问域名，`INSTANCE_ID` 为 FHIR 服务实例 ID。请在 [腾讯健康数据服务控制台](#) 的实例接入信息中获取，或以部署方提供的服务访问地址为准。

请求地址通常采用以下形式：

- 资源读写： `[baseUrl]/[resourceType]` 或 `[baseUrl]/[resourceType]/[id]`
- 历史版本读取： `[baseUrl]/[resourceType]/[id]/_history/[versionId]`
- 搜索： `[baseUrl]/[resourceType]?[searchParams]`
- 患者全量信息检索： `[baseUrl]/Patient/[id]/$everything`
- Bundle 事务/批处理： `[baseUrl]`

公共参数

鉴权请求公共参数

参数名称	类型	是否必填	说明
header.version	String	是	请求协议版本，示例为 <code>v0.1</code>
header.flag	Integer	是	请求标记，示例为 <code>0</code>
body.seq	Integer	是	请求序号，示例为 <code>0</code>
body.cmd	String	否	预留命令字段
body.token	String	否	预留令牌字段，获取 <code>AccessToken</code> 时通常为空
body.traceid	String	否	请求链路追踪标识
body.client.platform	Integer	否	客户端平台标识
body.client.env	String	否	调用环境标识
body.client.isTourist	Integer	否	游客标识
body.client.product	Integer	否	产品标识
body.payload.timestamp	Integer	是	毫秒级时间戳。服务端允许的时间窗口为请求到达服务端时间前后3600000毫秒（1小时），超出窗口会被拒绝

body.payload.apiKey	String	是	API 密钥
body.payload.signature	String	是	签名值
body.payload.scopes	Array	条件必填	申请的权限范围。开启知情同意访问控制时必填，且至少含 1 个 actor/<资源类型>/<ID>（如 "actor/Practitioner/P001"），否则访问 FHIR 接口返回 401（Claims 中缺少 scope 字段）。未开启时可留空（[]）
body.payload.context	Object	否	上下文扩展参数

FHIR 请求公共参数

参数名称	类型	是否必填	说明
Host	String	是	FHIR 服务访问域名，例如 HOSTNAME。请在 腾讯健康数据服务控制台 的实例接入信息中获取，或以部署方提供的服务访问地址为准
Authorization	String	是	访问令牌，格式为 Bearer <AccessToken>
Content-Type	String	否	请求体类型，写入类请求通常为 application/fhir+json；Patch 请求通常为 application/json-patch+json
Accept	String	否	响应格式，建议使用 application/fhir+json
resourceType	String	视接口而定	FHIR 资源类型，如 Patient、Observation、Encounter
id	String	视接口而定	资源唯一标识
searchParams	Query String	否	搜索参数，用于筛选结果集
versionId	String	否	历史版本号，用于 vRead 等场景

接口鉴权

平台采用基于 API Key、API Secret 和签名机制的令牌鉴权方式。调用方需先向鉴权服务申请 AccessToken，再携带该令牌访问 FHIR 接口。

鉴权流程

- 调用方持有服务对应的 `API Key` 和 `API Secret`。
- 调用方生成毫秒级时间戳 `timestamp`。
- 调用方使用签名算法生成 `signature`。
- 调用方将 `apiKey`、`timestamp`、`signature` 提交至鉴权接口。
- 鉴权服务校验签名和时间戳有效性。
- 校验通过后返回基于 JWT 的 `AccessToken`。
- 调用方在访问 FHIR Restful 接口时通过 `Authorization` 请求头携带该令牌。
- FHIR 服务收到请求后校验 `AccessToken`。
- 校验通过则返回业务结果，校验失败则拒绝访问。

签名算法

签名原文拼接方式：

```
message = apiKey + timestamp
```

签名算法：

```
signature = HMAC-SHA256(apiSecret, message)
```

❗ 说明：

- `message` 使用 UTF-8 编码。
- `timestamp` 为毫秒级时间戳。
- 签名结果使用十六进制字符串表示。
- 示例代码显式返回大写十六进制字符串；服务端进行校验时对签名大小写不敏感。

时间戳校验规则

服务端会校验请求时间戳是否在允许的超时时间窗口内。当前允许的时间窗口为请求到达服务端时间前后 `3600000` 毫秒（1 小时）：

- 若 `timestamp > 服务端当前时间 + 3600000`，请求将被拒绝。
- 若 `timestamp < 服务端当前时间 - 3600000`，请求将被拒绝。

典型失败情况如下：

- `signTime is timed out`
- `signature is invalid`

访问令牌使用方式

获取到 `AccessToken` 后，调用方需在访问 FHIR 接口时将其放入 HTTP Header:

```
Authorization: Bearer <AccessToken>
```

字段说明如下:

- Header Key: `Authorization`
- Header Value: `Bearer <空格><AccessToken>`

示例代码

签名获取 `access_token` 方法。

```
import hashlib
import hmac
import time
from typing import Optional

import requests

def sign(api_key: str, api_secret: str, sign_time: int) -> str:
    """签名

    Args:
        api_key: API密钥
        api_secret: API密钥对应的秘密
        sign_time: 签名时间（毫秒）

    Returns:
        签名字符串（大写十六进制）
    """
    message = f"{api_key}{sign_time}".encode("utf-8")
    h = hmac.new(api_secret.encode("utf-8"), message, hashlib.sha256)
    return h.hexdigest().upper()

# 校验签名逻辑（本地自测用）
```

```
def check_signature(api_key: str, api_secret: str, signature: str,
                    sign_time: int, timeout: int) -> Optional[str]:
    """校验签名

    Args:
        api_key: API密钥
        api_secret: API密钥对应的秘密
        signature: 签名字符串
        sign_time: 签名时间（毫秒）
        timeout: 超时时间（毫秒）

    Returns:
        None表示校验通过，否则返回错误信息
    """
    now_time = int(time.time() * 1000) # 当前时间（毫秒）

    if sign_time > now_time + timeout or sign_time < now_time - timeout:
        return "signTime is timed out"

    calculated_sign = sign(api_key, api_secret, sign_time)
    if signature.upper() != calculated_sign.upper():
        return "signature is invalid"

    return None


def request_auth_server(api_key: str, api_secret: str) -> Optional[str]:
    sign_time = int(time.time() * 1000)
    signature = sign(api_key, api_secret, sign_time)

    rsp = requests.post(
        # 将 <FHIR_SERVICE_HOST> 替换为实际鉴权服务访问域名
        url="https://<FHIR_SERVICE_HOST>/cmd/GetAccessToken",
        json={
            "header": {
                "version": "v0.1",
                "flag": 0,
            },
            "body": {
                "seq": 0,
```

```

        "cmd": "",
        "token": "",
        "traceid": "",
        "client": {
            "platform": 0,
            "env": "",
            "isTourist": 0,
            "product": 0,
        },
        "payload": {
            "timestamp": sign_time,
            "apiKey": api_key,
            "signature": signature,
            "scopes": [],
            "context": {},
        },
    },
    },
    timeout=10,
)

if rsp.status_code != 200:
    return None

data = rsp.json()
if data.get("retcode", 0) != 0:
    return None

payload = data.get("payload") or {}
return payload.get("accessToken")

if __name__ == "__main__":
    key = "<KEY>"
    secret = "<SECRET>"
    token = request_auth_server(key, secret)
    print(token)

```

返回结果

鉴权接口返回结果

鉴权接口调用成功后，返回结果中包含 `accessToken`，调用方应妥善保存并在后续 FHIR 请求中使用。返回结果示例如下：

```
{
  "payload": {
    "accessToken": "<ACCESS_TOKEN>"
  }
}
```

若鉴权接口调用失败，可能出现以下情况：

- HTTP 状态码非 `200`。
- 响应中未返回 `accessToken`。
- 签名校验失败或时间戳失效。

FHIR 接口返回结果

FHIR 接口返回结果遵循标准 FHIR 资源结构，不同接口场景返回内容不同：

- 资源创建、更新、读取类接口：通常返回对应资源内容。
- 搜索类接口：通常返回 `Bundle`。
- 删除类接口：通常返回 `OperationOutcome` 或状态结果。
- 事务/批处理类接口：通常返回结果 `Bundle`。

常见返回头包括：

参数名称	说明
Status Code	HTTP 状态码，用于标识请求处理结果
ETag	资源版本标识
Location / Content-Location	资源或历史版本访问地址

返回结果处理

- 对写入类接口，建议结合 `Status Code`、`ETag` 和 `Location` 判断是否写入成功。
- 对搜索类接口，建议优先解析 `Bundle.entry`、`Bundle.total` 和 `Bundle.link`。
- 对删除和异常场景，建议结合 `OperationOutcome` 或错误响应体定位具体原因。

资源基础操作接口

创建

最近更新时间：2026-07-28 16:18:01

接口描述

用于在 FHIR 数据服务中创建新的资源实例。该接口遵循标准 FHIR `Create` 操作，客户端通过 HTTP `POST` 方法向指定资源类型端点提交符合 FHIR 规范的资源内容，服务端完成资源落库后返回新建资源及其元数据。本示例展示了创建 `Patient` 资源的调用方式。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>POST</code>
URL	String	是	资源创建地址，格式为 <code>[baseUrl]/[resourceType]</code>
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）
Content-Type	String	否	请求体 MIME 类型，写入类请求通常为 <code>application/fhir+json</code> ；未指定时服务端将根据请求内容自动推断
Request Body	JSON Object	是	待创建的 FHIR 资源内容，需符合对应资源类型的 FHIR 结构定义

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir/Patient
```

请求体示例字段说明：

字段	类型	是否必填	说明
----	----	------	----

resourceType	String	是	FHIR 资源类型，本示例为 Patient 。当前实例实际支持的全部资源类型，请通过 GET /INSTANCE_ID/fhir/metadata 调用 CapabilityStatement 接口，在返回的 rest[].resource[].type 中查看（实例支持的资源类型可在创建时通过 supported_resource_types 配置裁剪，不同实例可能不同）。各资源类型的字段定义可参见 FHIR Resource Types
identifier	Array	否	患者标识信息
identifier[].system	String	否	标识体系，如 urn:oid:1.2.36.146.595.217.0.1
identifier[].value	String	否	标识值，如 12345
name	Array	否	姓名信息
name[].family	String	否	姓
name[].given	Array	否	名
gender	String	否	性别。合法取值为 male （男）、female （女）、other （其他）、unknown （未知），详情可参见 FHIR Patient.gender
birthDate	String	否	出生日期，格式为 YYYY-MM-DD

输出参数

接口调用成功后，返回 HTTP 状态码 201 Created ，并在响应头中包含新创建资源的位置及版本信息，响应体中返回创建后的完整资源内容。部分实例可能配置为创建时不返回响应体，此时可通过响应头中的 Location 地址调用 Read 接口获取完整资源内容。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 201 Created
ETag	String	资源版本标识，例如 W/"1"
Location	String	新建资源地址及历史版本地址

Content-Location	String	当前资源版本地址，格式为 <code>[baseUrl]/[resourceType]/[id]/_history/[versionId]</code>
x-request-id	String	请求追踪 ID

响应体主要字段说明：

字段	类型	说明
resourceType	String	资源类型
id	String	新创建资源的唯一 ID
meta.versionId	String	当前资源版本号
meta.lastUpdated	String	资源最后更新时间
meta.source	String	资源来源标识，格式为 <code>#<哈希值></code> ，用于标识资源的创建来源系统，可参见 FHIR Meta.source
meta.profile	Array	资源遵循的 FHIR StructureDefinition profile 列表
identifier	Array	资源标识信息
name	Array	资源姓名信息
gender	String	性别
birthDate	String	出生日期

说明：

`text` 为 FHIR 资源的 Narrative（人类可读摘要）字段，包含 `status` 和 `div` 两个子字段。服务端默认不自动生成 Narrative，响应体中通常不包含 `text` 字段；仅当客户端在请求体中主动提供 `text` 字段时，服务端会将其保留并返回。`text.status` 的合法取值为 `generated`（由系统生成）、`extensions`（仅包含扩展信息）和 `additional`（包含额外内容），详情请参见 [FHIR NarrativeStatus](#)。

示例

请求示例

```
POST /INSTANCE_ID/fhir/Patient HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Patient",
  "identifier": [
    {
      "system": "urn:oid:1.2.36.146.595.217.0.1",
      "value": "12345"
    }
  ],
  "name": [
    {
      "family": "Chalmers",
      "given": [
        "Peter",
        "James"
      ]
    }
  ],
  "gender": "male",
  "birthDate": "1974-12-25"
}
```

响应示例

```
HTTP/1.1 201 Created
ETag: W/"1"
Location: https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963/_history/1
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963/_history/1
x-request-id: ENqsVXqc5SeqLOfU
```

```
{
  "resourceType": "Patient",
```

```
"id": "199963",
"meta": {
  "versionId": "1",
  "lastUpdated": "2022-10-10T07:07:33.086-04:00",
  "source": "#oYcdS7hlfUBnsSlX",
  "profile": [
    "http://hl7.org/fhir/StructureDefinition/Patient"
  ]
},
"identifier": [
  {
    "system": "urn:oid:1.2.36.146.595.217.0.1",
    "value": "12345"
  }
],
"name": [
  {
    "family": "Chalmers",
    "given": [
      "Peter",
      "James"
    ]
  }
],
"gender": "male",
"birthDate": "1974-12-25"
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	请求参数错误，或提交的资源内容不符合 JSON/FHIR 结构要求
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无创建该资源的权限
404 Not Found	目标资源类型端点不存在

415 Unsupported Media Type	<code>Content-Type</code> 不受支持，例如未使用 <code>application/fhir+json</code>
422 Unprocessable Entity	资源内容通过语法校验但未通过业务或 FHIR 规则校验
500 Internal Server Error	服务端内部处理异常

更新

最近更新时间：2026-07-28 16:18:01

接口描述

用于更新指定资源 ID 对应的资源实例。该接口遵循标准 FHIR `Update` 操作，客户端通过 HTTP `PUT` 方法向指定资源地址提交完整资源内容，服务端更新成功后返回最新版本的资源数据及版本信息。

⚠ 注意：

`PUT` 更新为**全量替换**操作，提交的请求体将完整覆盖目标资源的现有内容。若需要在原有资源基础上新增或修改部分字段，请求体中必须包含资源的**全部字段**（已有字段 + 新增字段）；若仅提交部分字段，未包含的已有字段将被清除。如需仅更新局部字段而保留其余内容不变，请使用 [补丁](#) 接口（`PATCH` 方法 + `application/json-patch+json`）。

本示例展示了更新 `Patient` 资源的调用方式，在原有患者信息基础上新增地址信息（请求体中包含完整的患者信息及新增的地址字段）。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>PUT</code> 。
URL	String	是	资源更新地址，格式为 <code>[baseUrl]/[resourceType]/[id]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。
Content-Type	String	否	请求体 MIME 类型，写入类请求通常为 <code>application/fhir+json</code> ；未指定时服务端将根据请求内容自动推断。
Request Body	JSON Object	是	待更新的完整 FHIR 资源内容，需包含与 URL 一致的资源 ID。 <code>PUT</code> 为全量替换操作，请求体中未包含的字段将被清除。

请求 URL 示例：

https://HOSTNAME/INSTANCE_ID/fhir/Patient/id

请求体示例字段说明：

字段	类型	是否必填	说明
resourceType	String	是	FHIR 资源类型，本示例为 <code>Patient</code> 。当前实例实际支持的全部资源类型，请通过 <code>GET /INSTANCE_ID/fhir/metadata</code> 调用 <code>CapabilityStatement</code> 接口，在返回的 <code>rest[].resource[].type</code> 中查看（实例支持的资源类型可在创建时通过 <code>supported_resource_types</code> 配置裁剪，不同实例可能不同）。各资源类型的字段定义可参见 FHIR Resource Types 。
id	String	是	资源唯一 ID。为服务端创建资源时自动分配的逻辑 ID，可通过以下方式获取： - 调用 创建接口（POST）创建资源，从响应头 <code>Content-Location</code> 或响应体 <code>id</code> 字段获取。 - 调用 搜索接口（GET <code>/[resourceType]?_id=</code> 或 <code>/[resourceType]?identifier=</code>）查询已有资源 ID。
identifier	Array	否	患者标识信息。
name	Array	否	姓名信息。
gender	String	否	性别。合法取值为 <code>male</code> （男）、 <code>female</code> （女）、 <code>other</code> （其他）、 <code>unknown</code> （未知），详见 FHIR Patient.gender 。
birthDate	String	否	出生日期，格式为 <code>YYYY-MM-DD</code> 。
address	Array	否	地址信息。
address[].line	Array	否	街道地址。
address[].city	String	否	城市。

address[].state	String	否	省/州。
address[].postalCode	String	否	邮政编码。

输出参数

接口调用成功后，通常返回 HTTP 状态码 200 OK，并在响应头中包含资源版本信息与最新版本地址，响应体中返回更新后的完整资源内容。部分实例可能配置为更新时不返回响应体，此时可通过响应头中的 Content-Location 地址调用 Read 接口获取完整资源内容。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 200 OK。
ETag	String	资源版本标识，例如 W/"2"。
Content-Location	String	更新后资源的历史版本地址。
x-request-id	String	请求追踪 ID。

响应体主要字段说明：

字段	类型	说明
resourceType	String	资源类型。
id	String	被更新资源的唯一 ID。
meta.versionId	String	当前资源版本号。
meta.lastUpdated	String	资源最后更新时间。
meta.source	String	资源来源标识，格式为 #<哈希值>。仅在资源创建或更新时指定过该字段时才返回。可通过请求头 X-Source（需服务端启用 CaptureResourceSourceFromHeaderInterceptor）或客户端直接写入 meta.source 设置，详见 FHIR Meta.source。
meta.profile	Array	资源遵循的 FHIR StructureDefinition profile 列表。

identifier	Array	资源标识信息。
name	Array	资源姓名信息。
gender	String	性别。
birthDate	String	出生日期。
address	Array	更新后的地址信息。

说明：

`text` 为 FHIR 资源的 Narrative（人类可读摘要）字段，包含 `status` 和 `div` 两个子字段。服务端默认不自动生成 Narrative，响应体中通常不包含 `text` 字段；仅当客户端在请求体中主动提供 `text` 字段时，服务端会将其保留并返回。`text.status` 的合法取值为 `generated`（由系统生成）、`extensions`（仅包含扩展信息）和 `additional`（包含额外内容），详见 [FHIR NarrativeStatus](#)。

示例

请求示例

```
PUT /INSTANCE_ID/fhir/Patient/id HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Patient",
  "id": "id",
  "identifier": [
    {
      "system": "urn:oid:1.2.36.146.595.217.0.1",
      "value": "12345"
    }
  ],
  "name": [
    {
      "family": "Chalmers",
      "given": [
        "Peter",
        "James"
      ]
    }
  ]
}
```

```
    ]
  }
],
"gender": "male",
"birthDate": "1974-12-25",
"address": [
  {
    "line": [
      "534 Erewhon St"
    ],
    "city": "PleasantVille",
    "state": "Vic",
    "postalCode": "3999"
  }
]
```

响应示例

```
HTTP/1.1 200 OK
ETag: W/"2"
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/id/_history/2
x-request-id: faiRG8M5xyXBhbfv
```

```
{
  "resourceType": "Patient",
  "id": "id",
  "meta": {
    "versionId": "2",
    "lastUpdated": "2022-10-10T07:51:04.607-04:00",
    "source": "#oYcdS7hlfUBnsSlX",
    "profile": [
      "http://hl7.org/fhir/StructureDefinition/Patient"
    ]
  },
  "identifier": [
    {
```

```
      "system": "urn:oid:1.2.36.146.595.217.0.1",
      "value": "12345"
    }
  ],
  "name": [
    {
      "family": "Chalmers",
      "given": [
        "Peter",
        "James"
      ]
    }
  ],
  "gender": "male",
  "birthDate": "1974-12-25",
  "address": [
    {
      "line": [
        "534 Erewhon St"
      ],
      "city": "PleasantVille",
      "state": "Vic",
      "postalCode": "3999"
    }
  ]
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	请求参数错误，或提交的资源内容不符合 JSON/FHIR 结构要求
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无更新该资源的权限
404 Not Found	指定资源不存在
409 Conflict	资源状态冲突，例如版本冲突

415 Unsupported Media Type	<code>Content-Type</code> 不受支持，例如未使用 <code>application/fhir+json</code>
422 Unprocessable Entity	资源内容通过语法校验但未通过业务或 FHIR 规则校验
500 Internal Server Error	服务端内部处理异常

删除

最近更新时间：2026-07-28 16:18:01

接口描述

用于删除指定资源 ID 对应的资源实例。该接口遵循标准 FHIR `Delete` 操作，客户端通过 HTTP `DELETE` 方向指定资源地址发起请求。删除操作通常为逻辑删除，即资源会被标记为已删除，不再出现在常规检索结果中，但资源历史版本仍然保留。

逻辑删除通常具有以下语义：

- 资源被标记为已删除，不再出现在搜索结果中。
- 资源版本号会递增，并生成一个新的删除版本。
- 历史版本通常不会被物理删除。
- 在满足业务规则的前提下，资源后续可通过更新方式重新激活。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>DELETE</code> 。
URL	String	是	资源删除地址，格式为 <code>[baseUrl]/[resourceType]/[id]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， AccessToken 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。

路径参数说明：

字段	类型	是否必填	说明
resourceType	String	是	FHIR 资源类型，例如 <code>Patient</code> 、 <code>Observation</code> 、 <code>MedicationRequest</code> 等。当前实例实际支持的全部资源类型，请通过 <code>GET /INSTANCE_ID/fhir/metadata</code> 调用 <code>CapabilityStatement</code> 接口，在返回的 <code>rest[].resource[].type</code> 中查看（实例支持的资源类型可在创建时通过 <code>supported_resource_types</code> 配置裁剪，不同实例可能不同）。各资源类型的字段定义可参见 FHIR Resource Types 。

id	String	是	要删除的资源唯一 ID。为服务端创建资源时自动分配的逻辑 ID，可通过以下方式获取： - 调用 创建接口（POST）创建资源，从响应头 <code>Content-Location</code> 或响应体 <code>id</code> 字段获取。 - 调用 搜索接口（GET <code>/[resourceType]?_id=</code> 或 <code>/[resourceType]?identifier=</code>）查询已有资源 ID。
----	--------	---	---

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963
```

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，并在响应头中返回最新版本信息。响应体一般为 `OperationOutcome` 资源，用于描述删除执行结果。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 。
Content-Location	String	删除后资源对应的历史版本地址，格式为 <code>[baseUrl]/[resourceType]/[id]/_history/[versionId]</code> 。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>OperationOutcome</code> 。
issue	Array	执行结果明细。
issue[].severity	String	结果级别，如 <code>information</code> 。
issue[].code	String	结果代码，如 <code>informational</code> 。
issue[].diagnostics	String	删除结果描述信息，例如 <code>Successfully deleted 1 resource(s). Took 17ms.</code>

issue[].details	Object	结构化详细信息（ <code>CodeableConcept</code> 类型），包含 <code>coding</code> 等子字段，用于表达机器可读的错误详情。
-----------------	--------	--

示例

请求示例

```
DELETE /INSTANCE_ID/fhir/Patient/199963 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_JWT_TOKEN
```

响应示例

```
HTTP/1.1 200 OK
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963/_history/3
```

```
{
  "resourceType": "OperationOutcome",
  "issue": [
    {
      "severity": "information",
      "code": "informational",
      "details": {
        "coding": [
          {
            "system": "https://hapifhir.io/fhir/CodeSystem/hapi-fhir-storage-response-code",
            "code": "SUCCESSFUL_DELETE",
            "display": "Delete succeeded."
          }
        ]
      },
      "diagnostics": "Successfully deleted 1 resource(s). Took 17ms."
    }
  ]
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	请求格式错误或资源 ID 非法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无删除该资源的权限
404 Not Found	指定资源不存在
409 Conflict	当前资源状态不允许删除（如存在版本冲突导致无法删除）
410 Gone	资源已被逻辑删除或不可访问
422 Unprocessable Entity	资源违反业务规则或 FHIR 规范，无法删除
500 Internal Server Error	服务端内部处理异常

补丁

最近更新时间：2026-07-28 16:18:01

接口描述

用于对指定资源执行局部更新。该接口遵循标准 FHIR Patch 操作，客户端通过 HTTP PATCH 方法提交补丁文档，仅修改资源中的部分字段，无需传入完整资源内容。

本接口支持基于 JSON Patch 的更新方式，请求头 Content-Type 通常使用 application/json-patch+json。补丁文档以数组形式提交，每一项表示一个变更操作，常见操作包括：

- add：新增字段或新增数组元素。
- replace：替换现有字段值。
- remove：删除字段或数组元素。
- test：校验目标路径当前值是否符合预期，常用于执行删除前的条件检查。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 PATCH。
URL	String	是	资源局部更新地址，格式为 [baseUrl]/[resourceType]/[id]。
Authorization	String	是	身份认证令牌，格式为 Bearer <AccessToken>，AccessToken 通过 调用方式 中的 GetAccessToken 接口获取（实例控制台场景可通过 GetWebAccessToken 接口获取）。
Content-Type	String	是	请求体 MIME 类型，固定为 application/json-patch+json。
Request Body	JSON Array	是	JSON Patch 补丁文档，数组中的每一项代表一个操作。

路径参数说明：

字段	类型	是否必填	说明
----	----	------	----

resource Type	String	是	FHIR 资源类型，本示例为 Patient 。当前实例实际支持的全部资源类型，请通过 GET /INSTANCE_ID/fhir/metadata 调用 CapabilityStatement 接口，在返回的 rest[].resource[].type 中查看（实例支持的资源类型可在创建时通过 supported_resource_types 配置裁剪，不同实例可能不同）。各资源类型的字段定义可参见 FHIR Resource Types 。
id	String	是	资源唯一 ID。为服务端创建资源时自动分配的逻辑 ID，可通过以下方式获取： - 调用 创建接口 （POST）创建资源，从响应头 Content-Location 或响应体 id 字段获取。 - 调用 搜索接口 （GET /[resourceType]?_id= 或 /[resourceType]?identifier= ）查询已有资源 ID。

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir/Patient/149954
```

请求体示例字段说明：

字段	类型	是否必填	说明
op	String	是	补丁操作类型，如 add 、 replace 、 remove 、 test 。
path	String	是	变更目标路径，采用 JSON Pointer 格式，如 /gender 。
value	Any	否	变更值； add 、 replace 时通常必填。

- ❗ 说明：
- 补丁文档必须为数组，即使仅包含一个操作也需使用数组格式。
 - 对数组字段进行新增时，如目标数组不存在，通常需要先创建数组节点，再向指定下标写入内容。
 - path 必须准确指向资源中的目标字段，否则可能导致请求失败。
 - 删除数组元素时，需要明确待删除元素的数组下标，例如 /address/0 。
 - 如需避免误删，可先使用 test 操作校验目标元素内容，再执行 remove 操作。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，并在响应头中包含资源版本信息与最新版本地址，响应体中返回补丁执行后的最新资源内容。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 。
Content-Location	String	更新后资源的历史版本地址，格式为 <code>[baseUrl]/[resourceType]/[id]/_history/[versionId]</code> 。
Last-Modified	String	资源最后修改时间，格式为 HTTP-date，例如 <code>Mon, 06 Jul 2026 08:42:36 GMT</code> 。

响应体主要字段说明：

字段	类型	说明
resourceType	String	资源类型。
id	String	被更新资源的唯一 ID。
meta.versionId	String	当前资源版本号。
meta.lastUpdated	String	资源最后更新时间。
meta.source	String	资源来源标识，仅当资源创建/更新时指定过该字段时返回。
gender	String	更新后的性别字段（本示例为 Patient 资源）。
birthDate	String	更新后的出生日期（本示例为 Patient 资源）。
address	Array	更新后的地址信息（本示例为 Patient 资源）。

说明：

- 响应体中 `text` 字段（Narrative 摘要）是否返回取决于服务端 Narrative 配置。当配置开启（`narrative_enabled=true`）时返回 `text.status` 和 `text.div`；配置关闭时不返回 `text` 字段。
- `text.status` 可能值为 `generated`、`extensions`、`additional`、`empty`。
- `meta.source` 仅在资源创建或更新时指定过该字段时才返回。该字段用于标识资源的原始来源系统，可通过请求头 `X-Source`（需服务端启用 `CaptureResourceSourceFromHeaderIntercept` or）或客户端直接写入 `meta.source` 设置。

- 上表中的 `gender`、`birthDate`、`address` 是 Patient 资源示例字段，实际响应体中的业务字段取决于执行补丁的资源类型与修改内容。

示例

示例一：新增性别字段

请求示例

```
PATCH /INSTANCE_ID/fhir/Patient/149954 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <token>
Content-Type: application/json-patch+json
```

```
[
  {
    "op": "add",
    "path": "/gender",
    "value": "female"
  }
]
```

响应示例

```
HTTP/1.1 200 OK
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/149954/_history/2
Last-Modified: Mon, 06 Jul 2026 08:42:36 GMT
```

```
{
  "resourceType": "Patient",
  "id": "149954",
  "meta": {
    "versionId": "2",
    "lastUpdated": "2022-04-24T19:05:13.111+05:30"
  },
  "gender": "female"
```

```
}
```

示例二：新增地址信息

请求示例

```
PATCH /INSTANCE_ID/fhir/Patient/149954 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <token>
Content-Type: application/json-patch+json
```

```
[
  {
    "op": "add",
    "path": "/address",
    "value": []
  },
  {
    "op": "add",
    "path": "/address/0",
    "value": {
      "use": "home",
      "line": [
        "<Sample Street Name>",
        "avon"
      ],
      "city": "<City_Name>",
      "district": "<Sample Street Name>",
      "state": "Vic",
      "postalCode": "3999",
      "text": "<Sample Street Name>"
    }
  }
]
```

响应示例

```
HTTP/1.1 200 OK
```

```
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/149954/_history/3
Last-Modified: Mon, 06 Jul 2026 08:42:36 GMT
```

```
{
  "resourceType": "Patient",
  "id": "149954",
  "meta": {
    "versionId": "3",
    "lastUpdated": "2022-04-24T19:17:02.229+05:30"
  },
  "gender": "female",
  "address": [
    {
      "use": "home",
      "text": "<Sample Street Name>",
      "line": [
        "<Sample Street Name>",
        "avon"
      ],
      "city": "<City_Name>",
      "district": "<Sample Street Name>",
      "state": "Vic",
      "postalCode": "3999"
    }
  ]
}
```

示例三：替换邮编和出生日期

请求示例

```
PATCH /INSTANCE_ID/fhir/Patient/149954 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <token>
Content-Type: application/json-patch+json
```

```
[
```



```
{
  "op": "replace",
  "path": "/address/0/postalCode",
  "value": "4000"
},
{
  "op": "replace",
  "path": "/birthDate",
  "value": "1974-02-20"
}
]
```

响应示例

```
HTTP/1.1 200 OK
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/149954/_history/4
Last-Modified: Mon, 06 Jul 2026 08:42:36 GMT
```

```
{
  "resourceType": "Patient",
  "id": "149954",
  "meta": {
    "versionId": "4",
    "lastUpdated": "2022-04-24T19:37:51.559+05:30"
  },
  "gender": "female",
  "birthDate": "1974-02-20",
  "address": [
    {
      "use": "home",
      "text": "<Sample Street Name>",
      "line": [
        "<Sample Street Name>",
        "avon"
      ],
      "city": "<City_Name>",
      "district": "<Sample Street Name>",

```

```
    "state": "Vic",
    "postalCode": "4000"
  }
]
```

示例四：删除地址数组中的第一个元素

请求示例

```
PATCH /INSTANCE_ID/fhir/Patient/149954 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <token>
Content-Type: application/json-patch+json
```

```
[
  {
    "op": "remove",
    "path": "/address/0"
  }
]
```

响应示例

```
HTTP/1.1 200 OK
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/149954/_history/5
Last-Modified: Mon, 06 Jul 2026 08:42:36 GMT
```

```
{
  "resourceType": "Patient",
  "id": "149954",
  "meta": {
    "versionId": "5",
    "lastUpdated": "2022-04-24T19:54:03.769+05:30"
  },
  "gender": "female",
  "birthDate": "1974-02-20"
```

```
}
```

❗ 说明:

- 当目标字段为数组时，执行删除操作需要明确数组元素的位置。如果无法确保当前位置是否仍为目标数据，建议先使用 `test` 操作进行内容校验。
- 如果删除后数组变为空（即原数组只有 1 个元素），且实例开启了资源库校验（`enable_repository_validating_interceptor=true`），需要额外处理空数组和 `meta.source` 问题，请参见 [示例六](#)。

示例五：校验后删除地址信息

请求示例

```
PATCH /INSTANCE_ID/fhir/Patient/149954 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <token>
Content-Type: application/json-patch+json
```

```
[
  {
    "op": "test",
    "path": "/address/0",
    "value": {
      "use": "home",
      "line": [
        "<Sample Street Name>",
        "avon"
      ],
      "city": "<City_Name>",
      "district": "<Sample Street Name>",
      "state": "Vic",
      "postalCode": "4000",
      "text": "<Sample Street Name>"
    }
  },
  {
    "op": "remove",
    "path": "/address/0"
```

```
}  
]
```

响应示例

```
HTTP/1.1 200 OK  
Content-Location:  
https://HOSTNAME/INSTANCE_ID/fhir/Patient/149954/_history/5  
Last-Modified: Mon, 06 Jul 2026 08:42:36 GMT
```

```
{  
  "resourceType": "Patient",  
  "id": "149954",  
  "meta": {  
    "versionId": "5",  
    "lastUpdated": "2022-04-24T19:54:03.769+05:30"  
  },  
  "gender": "female",  
  "birthDate": "1974-02-20"  
}
```

示例六：开启数据校验时清空数组字段

请求示例

```
PATCH /INSTANCE_ID/fhir/Patient/149954 HTTP/1.1  
Host: HOSTNAME  
Authorization: Bearer <token>  
Content-Type: application/json-patch+json
```

```
[  
  {  
    "op": "replace",  
    "path": "/meta/source",  
    "value": "http://test.example.com/source"  
  },  
  {  
    "op": "remove",
```

```
    "path": "/address/0"
  },
  {
    "op": "remove",
    "path": "/address"
  }
]
```

响应示例

```
HTTP/1.1 200 OK
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/149954/_history/2
Last-Modified: Mon, 14 Jul 2026 12:17:58 GMT
```

```
{
  "resourceType": "Patient",
  "id": "149954",
  "meta": {
    "versionId": "2",
    "lastUpdated": "2026-07-14T20:17:58.710+08:00",
    "source": "http://test.example.com/source#rdVXtMxLI6QB74gC"
  },
  "identifier": [
    {
      "system": "http://test.example.com/patient-id",
      "value": "TEST-001"
    }
  ],
  "name": [
    {
      "family": "Test",
      "given": ["Validate"]
    }
  ],
  "gender": "male",
  "birthDate": "2000-01-01"
}
```

说明：

- 当实例开启了「资源库校验」（`enable_repository_validating_interceptor=true`，生产环境默认开启），PATCH 落地后的资源会经过 FHIR `RepositoryValidatingInterceptor` 全量校验。
- 第一条 `replace /meta/source` 把 `source` 改为有效 URL。HAPI 默认会将 `requestId` 拼成 `#requestId` 写入 `meta.source`，纯 `fragment` 不符合 URI 规范，需要替换为有效 URL。
- 第二条 `remove /address/0` 删除数组元素。
- 第三条 `remove /address` 删除空数组字段本身，避免留下空数组 `[]`。
- 如果实例未开启资源库校验，只需第二步 `remove` 即可（见示例四）。

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	补丁文档格式错误，或 <code>path</code> 、 <code>op</code> 参数非法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无更新该资源的权限
404 Not Found	指定资源不存在
409 Conflict	资源状态冲突，例如目标路径与当前资源状态不匹配
412 Precondition Failed	资源未通过 <code>RepositoryValidatingInterceptor</code> 数据校验。常见原因：PATCH 落地后出现空数组字段（如 <code>remove /address/0</code> 后未移除整个 <code>address</code> 数组）。
415 Unsupported Media Type	<code>Content-Type</code> 不受支持，例如未使用 <code>application/json-patch+json</code>
422 Unprocessable Entity	补丁语法正确，但未通过业务或 FHIR 规则校验
500 Internal Server Error	服务端内部处理异常

Read

最近更新时间：2026-07-28 16:18:01

接口描述

用于根据资源 ID 读取指定资源的最新版本内容。该接口遵循标准 FHIR `Read` 操作，客户端通过 HTTP `GET` 方法访问指定资源地址，服务端返回该资源当前最新版本的完整内容及版本信息。
本示例展示了读取 `Patient` 资源最新版本的调用方式。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	资源读取地址，格式为 <code>[baseUrl]/[resourceType]/[id]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。

路径参数说明：

字段	类型	是否必填	说明
resourceType	String	是	FHIR 资源类型，例如 <code>Patient</code> 、 <code>Observation</code> 、 <code>MedicationRequest</code> 等。当前实例实际支持的全部资源类型，请通过 <code>GET /INSTANCE_ID/fhir/metadata</code> 调用 <code>CapabilityStatement</code> 接口，在返回的 <code>rest[].resource[].type</code> 中查看（实例支持的资源类型可在创建时通过 <code>supported_resource_types</code> 配置裁剪，不同实例可能不同）。各资源类型的字段定义可参见 FHIR Resource Types 。
id	String	是	资源唯一 ID。为服务端创建资源时自动分配的逻辑 ID，可通过以下方式获取： - 调用 创建接口（POST）创建资源，从响应头 <code>Content-Location</code> 或响应体 <code>id</code> 字段获取。 - 调用 搜索接口（GET <code>/[resourceType]?_id=</code> 或 <code>/[resourceType]?identifier=</code>）查询已有资源 ID。

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963
```

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，并在响应头中包含资源版本信息与当前版本地址，响应体中返回资源完整内容。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 。
ETag	String	资源版本标识，例如 <code>W/"3"</code> 。
Content-Location	String	当前资源版本地址，格式为 <code>[baseUrl]/[resourceType]/[id]/_history/[versionId]</code> 。
Last-Modified	String	资源最后修改时间，格式为 HTTP-date，例如 <code>Fri, 03 Jul 2026 00:56:41 GMT</code> 。

响应体主要字段说明：

字段	类型	说明
resourceType	String	资源类型。
id	String	资源唯一 ID。
meta.versionId	String	当前资源版本号。
meta.lastUpdated	String	资源最后更新时间。
identifier	Array	标识信息。
name	Array	姓名信息。
gender	String	性别。
birthDate	String	出生日期。
address	Array	地址信息。

示例

请求示例

```
GET /INSTANCE_ID/fhir/Patient/199963 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
```

响应示例

```
HTTP/1.1 200 OK
ETag: W/"3"
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963/_history/3
Last-Modified: Mon, 06 Jul 2026 08:42:36 GMT
```

```
{
  "resourceType": "Patient",
  "id": "199963",
  "meta": {
    "versionId": "3",
    "lastUpdated": "2019-07-12T01:58:07.164+00:00"
  },
  "identifier": [
    {
      "system": "urn:oid:1.2.36.146.595.217.0.1",
      "value": "12345"
    }
  ],
  "name": [
    {
      "family": "Chalmers",
      "given": [
        "Peter",
        "James"
      ]
    }
  ]
},
```

```
"gender": "male",
"birthDate": "1974-02-13",
"address": [
  {
    "line": [
      "534 Erewhon St"
    ],
    "city": "PleasantVille",
    "state": "Vic",
    "postalCode": "M5C 2X8"
  }
]
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	请求格式错误或资源 ID 非法。
401 Unauthorized	未认证，缺少有效身份凭证。请检查请求头中是否携带正确的 <code>Authorization: Bearer <AccessToken></code> ，Token 是否已过期。
403 Forbidden	已认证但无读取该资源的权限。
404 Not Found	指定资源不存在。
410 Gone	资源已被逻辑删除或不可访问。
500 Internal Server Error	服务端内部处理异常。

vRead

最近更新时间：2026-07-28 16:18:01

接口描述

用于根据资源 ID 和版本号读取指定资源的历史版本内容。该接口遵循标准 FHIR `vRead` 操作，客户端通过 HTTP `GET` 方法访问指定资源的历史版本地址，服务端返回该资源对应版本的完整内容及版本信息。
本示例展示了读取 `Patient` 资源某一历史版本的调用方式。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	历史版本读取地址，格式为 <code>[baseUrl]/[resourceType]/[id]/_history/[versionId]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。

路径参数说明：

字段	类型	是否必填	说明
resourceType	String	是	FHIR 资源类型，例如 <code>Patient</code> 、 <code>Observation</code> 、 <code>MedicationRequest</code> 等。当前实例实际支持的全部资源类型，请通过 <code>GET /INSTANCE_ID/fhir/metadata</code> 调用 <code>CapabilityStatement</code> 接口，在返回的 <code>rest[].resource[].type</code> 中查看（实例支持的资源类型可在创建时通过 <code>supported_resource_types</code> 配置裁剪，不同实例可能不同）。各资源类型的字段定义可参见 FHIR Resource Types 。
id	String	是	资源唯一 ID。为服务端创建资源时自动分配的逻辑 ID，可通过以下方式获取： 调用 创建接口（POST）创建资源，从响应头 <code>Content-Location</code> 或响应体 <code>id</code> 字段获取。

			调用 搜索接口（GET <code>/[resourceType]?_id=</code> 或 <code>/[resourceType]?identifier=</code>）查询已有资源 ID。
version Id	String	是	资源历史版本号。可通过以下方式获取： - 调用 读取接口（GET <code>/[resourceType]/[id]</code>）时，响应头 <code>ETag</code> 或响应体 <code>meta.versionId</code> 字段即为当前版本号。 - 调用 历史接口（GET <code>/[resourceType]/[id]/_history</code>）可返回该资源的全部历史版本列表，其中每条记录的 <code>versionId</code> 即为可用版本号。

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963/_history/3
```

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，并在响应头中包含当前返回版本的版本信息与历史版本地址，响应体中返回对应版本的资源完整内容。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 。
ETag	String	资源版本标识，例如 <code>W/"3"</code> 。
Content-Location	String	当前返回的历史版本地址。
Last-Modified	String	资源最后修改时间，格式为 HTTP-date，例如 <code>Mon, 06 Jul 2026 08:42:36 GMT</code> 。

响应体主要字段说明：

字段	类型	说明
resourceType	String	资源类型。
id	String	资源唯一 ID。
meta.versionId	String	当前返回的资源版本号。
meta.lastUpdate	String	该版本资源的更新时间。

d		
meta.source	String	资源来源标识，仅当资源创建/更新时指定过该字段时返回。
identifier	Array	标识信息。
name	Array	姓名信息。
gender	String	性别。
birthDate	String	出生日期。
address	Array	地址信息。

- ❗ 说明：
- 响应体中 `text` 字段（Narrative 摘要）是否返回取决于服务端 Narrative 配置。当配置开启（`narrative_enabled=true`）时返回 `text.status` 和 `text.div`；配置关闭时不返回 `text` 字段。
 - `text.status` 可能值为 `generated`、`extensions`、`additional`、`empty`。
 - `meta.source` 仅在资源创建或更新时指定过该字段时才返回。该字段用于标识资源的原始来源系统，可通过请求头 `X-Source`（需服务端启用 `CaptureResourceSourceFromHeaderInterceptor`）或客户端直接写入 `meta.source` 设置。
 - 上表中的 `identifier`、`name`、`gender`、`birthDate`、`address` 是 Patient 资源示例字段，实际响应体中的业务字段取决于读取的资源类型。

示例

请求示例

```
GET /INSTANCE_ID/fhir/Patient/199963/_history/3 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <token>
```

响应示例

```
HTTP/1.1 200 OK
ETag: W/"3"
Content-Location:
https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963/_history/3
Last-Modified: Mon, 06 Jul 2026 08:42:36 GMT
```

```
{
  "resourceType": "Patient",
  "id": "199963",
  "meta": {
    "versionId": "3",
    "lastUpdated": "2019-07-12T01:58:07.164+00:00",
    "source": "#PR9lyCiz7HWynomw"
  },
  "identifier": [
    {
      "system": "urn:oid:1.2.36.146.595.217.0.1",
      "value": "12345"
    }
  ],
  "name": [
    {
      "family": "Chalmers",
      "given": [
        "Peter",
        "James"
      ]
    }
  ],
  "gender": "male",
  "birthDate": "1974-02-13",
  "address": [
    {
      "line": [
        "534 Erewhon St"
      ],
      "city": "PleasantVille",
      "state": "Vic",
      "postalCode": "M5C 2X8"
    }
  ]
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	请求格式错误，或版本号格式非法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无读取该资源历史版本的权限
404 Not Found	指定资源或历史版本不存在
410 Gone	指定历史版本已不可访问
500 Internal Server Error	服务端内部处理异常

搜索操作接口

基础搜索接口

最近更新时间：2026-07-28 16:18:01

接口描述

用于基于标准 FHIR 搜索能力对指定资源类型进行条件检索。客户端可通过 HTTP GET 方法访问资源类型端点，并在 URL 中携带搜索参数，以筛选符合条件的资源记录。

基础搜索支持以下常见方式：

- 无参数搜索：返回指定资源类型的记录列表。
- 单参数搜索：基于单个搜索字段筛选结果。
- 多参数组合搜索：多个参数之间默认按 AND 关系组合。
- 逗号分隔值搜索：同一参数中多个值按 OR 关系匹配。

本示例展示了对 Patient 资源执行基础搜索的常见方式。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 GET 。
URL	String	是	搜索地址，格式为 [baseUrl]/[resourceType] 或 [baseUrl]/[resourceType]?[searchParams] 。
Authorization	String	是	访问令牌，格式为 Bearer <AccessToken> 。获取方式详见 调用方式 。
Accept	String	否	响应格式，建议使用 application/fhir+json 。
searchParams	String	否	搜索参数，以 URL 查询字符串（ key=value&... ）形式拼接在 URL 之后，支持单个或多个参数组合。

常用搜索参数说明：

参数名称	类型	是否必填	说明
name	String	否	按患者姓名检索，可匹配姓名相关字段。

gender	String	否	按性别检索，如 <code>male</code> 、 <code>female</code> 。
family	String	否	按姓氏检索。
given	String	否	按名字检索；支持逗号分隔多个值。

- ❗ 说明：
- 当不传任何搜索参数时，系统返回指定资源类型的记录列表。
 - 多个不同参数同时传入时，默认按 AND 关系组合。
 - 同一参数传入逗号分隔多个值时，通常按 OR 关系处理。
 - 搜索结果通常按分页策略返回，单次响应可能只包含部分记录。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，响应体一般为 `Bundle` 资源，其中包含本次检索命中的资源列表及分页信息。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>Bundle</code> 。
id	String	<code>Bundle</code> 资源的唯一 ID，通常为 UUID。
meta	Object	<code>Bundle</code> 元数据，包含 <code>lastUpdated</code> 等字段。
meta.lastUpdated	String	<code>Bundle</code> 生成时间，ISO 8601格式。
type	String	<code>Bundle</code> 类型，搜索场景通常为 <code>searchset</code> 。
total	Integer	命中的总记录数。默认情况下可能不返回此字段，仅当服务端能够高效计算总数时返回；当前服务端不支持通过 <code>_total=accurate</code> 强制返回准确总数。
entry	Array	搜索结果列表。
entry[].fullUrl	String	资源完整访问地址。
entry[].resource	Object	匹配到的资源内容。

entry[].resource.id	String	资源的唯一 ID。
entry[].resource.meta	Object	资源元数据，包含 <code>versionId</code> 、 <code>lastUpdated</code> 等。
entry[].search.mode	String	搜索命中模式。
link	Array	分页链接信息，如首页、下一页等。

示例

请求示例

- 示例一：无参数搜索

```
GET /INSTANCE_ID/fhir/Patient HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例二：按姓名搜索

```
GET /INSTANCE_ID/fhir/Patient?name=chalmers HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例三：多参数组合搜索（AND）

```
GET /INSTANCE_ID/fhir/Patient?name=chalmers&gender=male HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例四：逗号分隔多值搜索（OR）

```
GET /INSTANCE_ID/fhir/Patient?given=peter,james HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
```

```
Accept: application/fhir+json
```

响应示例

```
{
  "resourceType": "Bundle",
  "id": "2d7b3566-1773-40b3-abd4-0c8ebdd6466f",
  "meta": {
    "lastUpdated": "2026-07-08T21:24:05.941+08:00"
  },
  "type": "searchset",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir/Patient?name=chalmers"
    }
  ],
  "entry": [
    {
      "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963",
      "resource": {
        "resourceType": "Patient",
        "id": "199963",
        "meta": {
          "versionId": "1",
          "lastUpdated": "2022-10-10T07:07:33.086-04:00"
        },
        "name": [
          {
            "family": "Chalmers",
            "given": [
              "Peter",
              "James"
            ]
          }
        ],
        "gender": "male"
      },
      "search": {
```

```
      "mode": "match"
    }
  }
]
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	搜索参数格式错误，或参数值不符合要求
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无搜索该资源的权限
404 Not Found	指定资源类型不存在
422 Unprocessable Entity	搜索参数语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

引用搜索接口

最近更新时间：2026-07-28 16:18:01

接口描述

用于基于资源之间的引用关系执行检索。当某个资源字段引用了另一类资源时，可通过引用字段作为搜索参数，筛选与目标资源相关联的数据记录。

本示例展示了基于 `subject` 引用字段检索 `Encounter` 资源的方式，即查询指定患者关联的就诊记录，并通过 `_sort` 参数对结果按日期升序排列。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	搜索地址，格式为 <code>[baseUrl]/[resourceType]?[searchParams]</code> 。
Authorization	String	是	访问令牌，格式为 <code>Bearer <AccessToken></code> 。获取方式详见 调用方式 。
Accept	String	否	响应格式，建议使用 <code>application/fhir+json</code> 。
subject	String	是	引用搜索参数，表示关联的患者资源引用，格式通常为 <code>Patient/{id}</code> 。其中 <code>{id}</code> 为被引用资源的唯一标识，可通过 基础搜索接口 或控制台资源查看器获取。
_sort	String	否	排序参数，用于指定结果排序字段。本示例按 <code>date</code> 排序。
resourceType	String	是	FHIR 资源类型，支持 FHIR 标准资源类型，如 <code>Patient</code> 、 <code>Encounter</code> 、 <code>Observation</code> 等，本示例为 <code>Encounter</code> 。

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir/Encounter?subject=Patient/k62d9d82-3f2b-44db-b808-04159be709fd&_sort=date
```

说明：

- 引用搜索适用于存在资源关联关系的场景。
- `subject=Patient/{id}` 表示仅返回关联到该患者的 `Encounter` 记录。
- `_sort=date` 表示按日期字段排序，示例中为从旧到新返回。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，响应体一般为 `Bundle` 资源，包含符合条件的 `Encounter` 记录列表及分页信息。

响应体主要字段说明：

字段	类型	说明
<code>resourceType</code>	<code>String</code>	返回资源类型，通常为 <code>Bundle</code> 。
<code>id</code>	<code>String</code>	<code>Bundle</code> 资源的唯一 ID，通常为 UUID。
<code>meta</code>	<code>Object</code>	<code>Bundle</code> 元数据，包含 <code>lastUpdated</code> 等字段。
<code>meta.lastUpdated</code>	<code>String</code>	<code>Bundle</code> 生成时间，ISO 8601 格式。
<code>type</code>	<code>String</code>	<code>Bundle</code> 类型，搜索场景通常为 <code>searchset</code> 。
<code>total</code>	<code>Integer</code>	命中的总记录数。默认情况下可能不返回此字段，仅当服务端能够高效计算总数时返回；当前服务端不支持通过 <code>_total=accurate</code> 强制返回准确总数。
<code>entry</code>	<code>Array</code>	搜索结果列表。
<code>entry[].fullUrl</code>	<code>String</code>	资源完整访问地址。
<code>entry[].resource</code>	<code>Object</code>	匹配到的 <code>Encounter</code> 资源内容。
<code>entry[].resource.id</code>	<code>String</code>	资源的唯一 ID。
<code>entry[].resource.meta</code>	<code>Object</code>	资源元数据，包含 <code>versionId</code> 、 <code>lastUpdated</code> 等。
<code>entry[].resource.subject</code>	<code>Object</code>	当前就诊记录关联的患者引用信息。
<code>entry[].resource.</code>	<code>Object</code>	就诊时间范围信息。

period		
link	Array	分页链接信息。

示例

请求示例

```
GET /INSTANCE_ID/fhir/Encounter?subject=Patient/k62d9d82-3f2b-44db-b808-04159be709fd&_sort=date HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

响应示例

```
{
  "resourceType": "Bundle",
  "id": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
  "meta": {
    "lastUpdated": "2024-01-10T08:00:00.000+08:00"
  },
  "type": "searchset",
  "entry": [
    {
      "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Encounter/enc-10001",
      "resource": {
        "resourceType": "Encounter",
        "id": "enc-10001",
        "meta": {
          "versionId": "1",
          "lastUpdated": "2024-01-10T08:00:00.000+08:00"
        },
        "subject": {
          "reference": "Patient/k62d9d82-3f2b-44db-b808-04159be709fd"
        },
        "period": {
          "start": "2024-01-10T08:00:00+08:00",
          "end": "2024-01-10T09:30:00+08:00"
        }
      }
    }
  ]
}
```

```
        }  
      }  
    }  
  ]  
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	搜索参数格式错误，或引用参数值不合法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无搜索该资源的权限
404 Not Found	指定资源类型不存在
422 Unprocessable Entity	搜索参数语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

数值搜索接口

最近更新时间：2026-07-28 16:18:01

接口描述

用于基于数值类字段对资源进行条件检索，常见于检验结果、生命体征、测量值等场景。客户端可通过 HTTP `GET` 方法访问目标资源类型端点，并结合标准搜索参数对检验项目编码、数值大小、单位等条件进行筛选。

本示例展示了基于 `Observation` 资源进行实验室检验结果搜索的几种常见方式，包括：

- 查询指定患者的全部实验室检验结果。
- 查询所有患者的某一项特定检验结果。
- 查询满足特定数值范围的检验结果。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	搜索地址，格式为 <code>[baseUrl]/[resourceType]?[searchParams]</code> 。
Authorization	String	是	访问令牌，格式为 <code>Bearer <AccessToken></code> 。获取方式详见 调用方式 。
Accept	String	否	响应格式，建议使用 <code>application/fhir+json</code> 。
subject	String	否	按关联患者过滤，格式通常为 <code>Patient/{id}</code> 。其中 <code>{id}</code> 为被引用资源的唯一标识，可通过 基础搜索接口 或控制台资源查看器获取。
category	String	否	按资源分类过滤，本示例为实验室分类，如 <code>http://hl7.org/fhir/observation-category%7Claboratory</code> 。
code	String	否	按检验项目编码过滤，通常为标准术语编码，如 <code>http://loinc.org%7C6298-4</code> 。注意 URL 中 <code>system%7Ccode</code> 的竖线 <code> </code> 需编码为 <code>%7C</code> 。
value-quantity	String	否	按数值和单位过滤，格式通常为 <code>[prefix][value]%7C[system]%7C[unit]</code> ，其中 <code>prefix</code> 为比较前缀（ <code>lt</code> 、 <code>le</code> 、 <code>g</code>

t、ge 等)，竖线 | 需编码为 %7C。

说明：

- category 可用于限制仅返回实验室类 Observation 资源。
- code 常用于检索某个具体的检验项目，例如钾离子检验。
- value-quantity 支持比较前缀，如 lt、le、gt、ge 等。
- 单位过滤通常建议同时带上单位系统和单位编码，以提高匹配准确性。
- FHIR 搜索参数中用于分隔 system 与 code 的竖线 \ | 在 URL 中需编码为 %7C，否则可能导致请求失败。

输出参数

接口调用成功后，通常返回 HTTP 状态码 200 OK，响应体一般为 Bundle 资源，包含符合条件的 Observation 资源列表及分页信息。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 Bundle。
id	String	Bundle 资源的唯一 ID，通常为 UUID。
meta	Object	Bundle 元数据，包含 lastUpdated 等字段。
meta.lastUpdated	String	Bundle 生成时间，ISO 8601 格式。
type	String	Bundle 类型，搜索场景通常为 searchset。
link	Array	分页链接信息，包含 relation 与 url。
link[].relation	String	链接关系类型，如 self、next、previous。
link[].url	String	对应分页的完整请求地址。
entry	Array	搜索结果列表。
entry[].fullUrl	String	资源的完整访问地址。
entry[].resource	Object	匹配到的 Observation 资源内容。
entry[].resource.id	String	资源的唯一 ID。

entry[].resource.metadata	Object	资源元数据，包含 <code>versionId</code> 、 <code>lastUpdated</code> 、 <code>source</code> 等。
entry[].resource.status	String	观察结果状态，如 <code>final</code> 、 <code>preliminary</code> 等。
entry[].resource.category	Array	资源分类信息，如实验室分类。
entry[].resource.code	Object	检验项目编码信息。
entry[].resource.subject	Object	关联患者引用信息。
entry[].resource.valueQuantity	Object	检验数值及单位信息。
entry[].search.mode	String	搜索命中模式，如 <code>match</code> 。

示例

请求示例

- 示例一：查询指定患者的全部实验室检验结果。

```
GET /INSTANCE_ID/fhir/Observation?subject=Patient/h0e270d9-ed56-4041-9dec-a45a73461c66&category=http://hl7.org/fhir/observation-category%7Claboratory HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例二：查询所有患者的某项特定检验结果。

```
GET /INSTANCE_ID/fhir/Observation?code=http://loinc.org%7C6298-4 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例三：查询数值低于 4.0 mmol/L 的钾离子结果。

```
GET /INSTANCE_ID/fhir/Observation?code=http://loinc.org%7C6298-4&value-quantity=lt4.0%7Chttp://unitsofmeasure.org%7Cmmol/L HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

响应示例

```
{
  "resourceType": "Bundle",
  "id": "a9fdb12d-8a08-4cfb-8d8c-13f8a0fa50ad",
  "meta": {
    "lastUpdated": "2026-07-08T21:24:14.191+08:00"
  },
  "type": "searchset",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir/Observation?code=http%3A%2F%2Floinc.org%7C6298-4&value-quantity=lt4.0%7Chttp%3A%2F%2Funitsofmeasure.org%7Cmmol%2FL"
    }
  ],
  "entry": [
    {
      "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Observation/9a6adc7c-2496-43b9-881f-72a3f3067544",
      "resource": {
        "resourceType": "Observation",
        "id": "9a6adc7c-2496-43b9-881f-72a3f3067544",
        "meta": {
          "versionId": "1",
          "lastUpdated": "2026-07-08T21:24:13.487+08:00",
          "source": "#nB7pryS0ElzarOdc"
        },
        "status": "final",
        "category": [
          {

```

```
      "coding": [
        {
          "system": "http://hl7.org/fhir/observation-category",
          "code": "laboratory"
        }
      ]
    },
    "code": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "6298-4",
          "display": "Potassium"
        }
      ]
    },
    "subject": {
      "reference": "Patient/f25a8f0c-3473-4deb-8917-45d967263a1e"
    },
    "valueQuantity": {
      "value": 3.8,
      "system": "http://unitsofmeasure.org",
      "code": "mmol/L"
    }
  },
  "search": {
    "mode": "match"
  }
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	搜索参数格式错误，或数值/单位表达式不合法

401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无搜索该资源的权限
404 Not Found	指定资源类型不存在
422 Unprocessable Entity	搜索参数语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

日期时间搜索接口

最近更新时间：2026-07-28 16:18:01

接口描述

用于基于日期或时间字段对资源进行条件检索。客户端可通过 HTTP `GET` 方法，在 URL 中传入日期类搜索参数，以筛选满足指定日期、年份、时间范围或更新时间条件的资源记录。

本示例展示了常见的日期时间搜索场景，包括：

- 按指定生日检索患者。
- 按指定年份检索患者。
- 按日期范围检索就诊记录。
- 按资源最后更新时间检索数据。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	搜索地址，格式为 <code>[baseUrl]/[resourceType]?[searchParameters]</code> 。
Authorization	String	是	访问令牌，格式为 <code>Bearer <AccessToken></code> 。获取方式详见 调用方式 。
Accept	String	否	响应格式，建议使用 <code>application/fhir+json</code> 。
birthdate	String	否	按出生日期检索，支持完整日期（如 <code>1974-02-13</code> ）或部分日期（如仅传年份 <code>1974</code> ）。
date	String	否	按日期字段检索，可与 <code>ge</code> 、 <code>le</code> 等比较前缀组合。
_lastUpdated	String	否	按资源最后更新时间检索，取值通常为 ISO 8601 日期时间。
subject	String	否	引用搜索参数，可与日期参数组合使用，如 <code>Patient/{id}</code> 。

说明：

- 日期参数支持完整日期，也支持部分日期，例如仅传年份。
- 日期范围通常通过多个带比较前缀的同名参数组合实现。
- `_lastUpdated` 可用于任意资源类型，筛选在指定时间点之后或之前更新的数据。
- 常见比较前缀包括 `ge`、`gt`、`le`、`lt`、`eq` 等。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，响应体一般为 `Bundle` 资源，其中包含符合条件的资源列表及分页信息。

响应体主要字段说明：

字段	类型	说明
<code>resourceType</code>	<code>String</code>	返回资源类型，通常为 <code>Bundle</code> 。
<code>id</code>	<code>String</code>	<code>Bundle</code> 资源的唯一 ID，通常为 UUID。
<code>meta</code>	<code>Object</code>	<code>Bundle</code> 元数据，包含 <code>lastUpdated</code> 等字段。
<code>meta.lastUpdated</code>	<code>String</code>	<code>Bundle</code> 生成时间。
<code>type</code>	<code>String</code>	<code>Bundle</code> 类型，搜索场景通常为 <code>searchset</code> 。
<code>total</code>	<code>Integer</code>	命中的总记录数。仅在请求中传入 <code>_total=accurate</code> 时返回；未传该参数时可能不包含此字段。
<code>link</code>	<code>Array</code>	分页链接信息，包含 <code>relation</code> 与 <code>url</code> 。
<code>link[].relation</code>	<code>String</code>	链接关系类型，如 <code>self</code> 、 <code>next</code> 、 <code>previous</code> 。
<code>link[].url</code>	<code>String</code>	对应分页的完整请求地址。
<code>entry</code>	<code>Array</code>	搜索结果列表。
<code>entry[].fullUrl</code>	<code>String</code>	资源的完整访问地址。
<code>entry[].resource</code>	<code>Object</code>	匹配到的资源内容。
<code>entry[].resource.id</code>	<code>String</code>	资源的唯一 ID。
<code>entry[].resource</code>	<code>Object</code>	资源元数据，包含 <code>versionId</code> 、 <code>lastUpdated</code> 等。

.meta		
entry[].search.mode	String	搜索命中模式，如 <code>match</code> 。

❗ 说明：

`total` 字段默认可能不返回。若需获取命中总数，请在请求参数中传入 `_total=accurate`；此时服务端会统计并返回 `total`。未传入 `_total` 参数时，响应体中可能不包含 `total` 字段。

示例

请求示例

- 示例一：查询生日为指定日期的患者。

```
GET /INSTANCE_ID/fhir/Patient?birthdate=1974-02-13 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例二：查询出生于指定年份的患者。

```
GET /INSTANCE_ID/fhir/Patient?birthdate=1974 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例三：查询指定日期范围内的就诊记录。

```
GET /INSTANCE_ID/fhir/Encounter?subject=Patient/ef2c19c4-ea06-473d-8781-368e4441c5c0&date=ge2009-06-01&date=le2009-07-31 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例四：查询在指定时间点之后更新的患者资源。

```
GET /INSTANCE_ID/fhir/Patient?_lastUpdated=ge2017-01-01T00:00:00Z
HTTP/1.1
```

```
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

响应示例

```
{
  "resourceType": "Bundle",
  "id": "180f7368-790e-4f2b-b24a-550eb027d747",
  "meta": {
    "lastUpdated": "2022-10-10T07:07:33.086-04:00"
  },
  "type": "searchset",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir/Patient?birthdate=1974-02-13"
    }
  ],
  "entry": [
    {
      "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963",
      "resource": {
        "resourceType": "Patient",
        "id": "199963",
        "meta": {
          "versionId": "1",
          "lastUpdated": "2022-10-10T07:07:33.086-04:00"
        },
        "birthDate": "1974-02-13"
      },
      "search": {
        "mode": "match"
      }
    }
  ]
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	日期参数格式错误，或比较前缀使用不合法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无搜索该资源的权限
404 Not Found	指定资源类型不存在
422 Unprocessable Entity	搜索参数语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

分页与结果数量控制

最近更新时间：2026-07-28 16:18:01

接口描述

用于对 FHIR 搜索结果进行分页获取及返回条数控制。客户端可通过 HTTP `GET` 方法发起搜索请求，服务端通常会以分页方式返回结果集合，并在响应的 `Bundle.link` 中提供后续分页链接。同时，也可通过 `_count` 参数控制单页返回的记录数。

本示例展示了以下两类常见场景：

- 搜索结果分页返回。
- 使用 `_count` 控制单次返回条数。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	搜索地址，格式为 <code>[baseUrl]/[resourceType]?[searchParams]</code> 。
Authorization	String	是	访问令牌，格式为 <code>Bearer <AccessToken></code> 。获取方式详见 调用方式 。
Accept	String	否	响应格式，建议使用 <code>application/fhir+json</code> 。
<code>_count</code>	Integer	否	指定单页返回记录数，默认值为 <code>20</code> ，需为正整数。实际最大值受服务端配置约束。
其他搜索参数	Query String	否	可与分页参数组合使用的常规搜索条件。

- ❗ 说明：
- 搜索结果通常默认分页返回。
 - 第一页结果中通常会提供 `self` 和 `next` 等分页链接。
 - 存在更多结果时，可通过 `next` 链接继续获取后续页面。
 - `_count` 仅控制单页返回条数，不改变命中的总结果数。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，响应体一般为 `Bundle` 资源，包含当前页结果、总数以及分页链接信息。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>Bundle</code> 。
id	String	当前结果集唯一标识。
meta.lastUpdated	String	当前结果集生成时间。
type	String	<code>Bundle</code> 类型，搜索场景通常为 <code>searchset</code> 。
total	Integer	命中的总记录数。默认情况下可能不返回此字段，仅当服务端能够高效计算总数时返回；当前服务端不支持通过 <code>_total=accurate</code> 强制返回准确总数。
link	Array	分页链接信息。
link[].relation	String	链接关系，如 <code>self</code> 、 <code>next</code> 。
link[].url	String	对应分页访问地址。
entry	Array	当前页返回的资源列表。

示例

示例一：分页查询搜索结果

请求示例

```
GET /INSTANCE_ID/fhir/Patient?birthdate=1974-02-13 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

响应示例

```
{
```

```
"resourceType": "Bundle",
"id": "3c4b5bfc-a03e-474c-9cb2-bd640022cf20",
"meta": {
  "lastUpdated": "2022-10-13T14:07:20.613+00:00"
},
"type": "searchset",
"link": [
  {
    "relation": "self",
    "url": "https://HOSTNAME/INSTANCE_ID/fhir/Patient?birthdate=1974-02-13"
  },
  {
    "relation": "next",
    "url": "https://HOSTNAME/INSTANCE_ID/fhir?_getpages=3c4b5bfc-a03e-474c-9cb2-bd640022cf20&_getpagesoffset=20&_count=20&_bundletype=searchset"
  }
]
```

示例二：控制单次返回结果数量

请求示例

```
GET /INSTANCE_ID/fhir/Observation?code=http://loinc.org%7C718-7&_count=20 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

响应示例

```
HTTP/1.1 200 OK
Content-Type: application/fhir+json
x-request-id: ENqsVXqc5SeqLOfU
```

```
{
```

```

"resourceType": "Bundle",
"id": "3c4b5bfc-a03e-474c-9cb2-bd640022cf20",
"meta": {
  "lastUpdated": "2022-10-13T14:07:20.613+00:00"
},
"type": "searchset",
"total": 45,
"link": [
  {
    "relation": "self",
    "url": "https://HOSTNAME/INSTANCE_ID/fhir/Observation?code=http://loinc.org%7C718-7&_count=20"
  },
  {
    "relation": "next",
    "url": "https://HOSTNAME/INSTANCE_ID/fhir?_getpages=3c4b5bfc-a03e-474c-9cb2-bd640022cf20&_getpagesoffset=20&_count=20&_bundletype=searchset"
  }
],
"entry": [
  {
    "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Observation/OBS001",
    "resource": {
      "resourceType": "Observation",
      "id": "OBS001",
      "status": "final",
      "code": {
        "coding": [
          {
            "system": "http://loinc.org",
            "code": "718-7",
            "display": "Hemoglobin [Mass/volume] in Blood"
          }
        ]
      }
    }
  },
  {
    "search": {
      "mode": "match"
    }
  }
]

```

```
}  
]  
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	分页参数或 <code>_count</code> 参数格式错误
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无搜索该资源的权限
404 Not Found	指定资源类型不存在
422 Unprocessable Entity	搜索参数语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

排序搜索

最近更新时间：2026-07-28 16:18:01

接口描述

用于对 FHIR 搜索结果按指定字段进行排序。客户端可通过 HTTP GET 方法发起搜索请求，并使用 `_sort` 参数指定排序字段，从而按业务需要调整结果顺序。排序搜索支持按单个搜索字段排序和按多个搜索字段组合排序。本示例展示了对 `Patient` 资源按姓氏和名字进行排序的常见方式。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	搜索地址，格式为 <code>[baseUrl]/[resourceType]?[searchParams]</code> 。
Authorization	String	是	访问令牌，格式为 <code>Bearer <AccessToken></code> 。获取方式详见 调用方式 。
Accept	String	否	响应格式，建议使用 <code>application/fhir+json</code> 。
<code>_sort</code>	String	是	排序字段名称，可指定一个或多个字段，多个字段之间使用逗号分隔。

- ❗ 说明：
- `_sort=family` 表示按姓氏排序。
 - `_sort=family,given` 表示先按姓氏排序，再按名字排序。
 - 排序字段通常需要为该资源类型支持的搜索参数。
 - 排序方向如需区分升序或降序，应以平台实际支持规则为准。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，响应体一般为 `Bundle` 资源，包含排序后的资源列表及分页信息。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>Bundle</code> 。
id	String	当前结果集唯一标识。
meta.lastUpdated	String	当前结果集生成时间。
type	String	<code>Bundle</code> 类型，搜索场景通常为 <code>searchset</code> 。
total	Integer	命中的总记录数。默认情况下可能不返回此字段，仅当服务端能够高效计算总数时返回；当前服务端不支持通过 <code>_total=accurate</code> 强制返回准确总数。
entry	Array	排序后的结果列表。
entry[].fullUrl	String	资源完整访问地址。
entry[].resource	Object	匹配到的资源内容。
entry[].resource.id	String	资源的唯一 ID。
entry[].resource.meta	Object	资源元数据，包含 <code>versionId</code> 、 <code>lastUpdated</code> 等。
entry[].search.mode	String	搜索命中模式。
link	Array	分页链接信息。

示例

请求示例

- 示例一：按单个字段排序。

```
GET /INSTANCE_ID/fhir/Patient?_sort=family HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例二：按多个字段排序。

```
GET /INSTANCE_ID/fhir/Patient?_sort=family,given HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

响应示例

```
{
  "resourceType": "Bundle",
  "id": "2d7b3566-1773-40b3-abd4-0c8ebdd6466f",
  "meta": {
    "lastUpdated": "2026-07-08T21:24:05.941+08:00"
  },
  "type": "searchset",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir/Patient?_sort=family"
    },
    {
      "relation": "next",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir?_getpages=2d7b3566-1773-40b3-abd4-0c8ebdd6466f&_getpagesoffset=2&_count=20&_bundletype=searchset"
    }
  ],
  "entry": [
    {
      "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Patient/199963",
      "resource": {
        "resourceType": "Patient",
        "id": "199963",
        "meta": {
          "versionId": "1",
          "lastUpdated": "2022-10-10T07:07:33.086-04:00"
        },
        "name": [
          {
            "family": "Chalmers",
```

```
        "given": [
            "James"
        ]
    },
]
},
"search": {
    "mode": "match"
}
},
{
    "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Patient/199964",
    "resource": {
        "resourceType": "Patient",
        "id": "199964",
        "meta": {
            "versionId": "1",
            "lastUpdated": "2022-10-10T07:07:33.086-04:00"
        },
        "name": [
            {
                "family": "Chalmers",
                "given": [
                    "Peter"
                ]
            }
        ]
    },
    "search": {
        "mode": "match"
    }
}
]
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
-----	----

400 Bad Request	<code>_sort</code> 参数格式错误，或排序字段不合法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无搜索该资源的权限
404 Not Found	指定资源类型不存在
422 Unprocessable Entity	排序参数语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

患者全量信息检索

最近更新时间：2026-07-28 16:18:01

接口描述

用于检索与指定患者相关联的全部资源数据。该接口通过患者实例上的 `$everything` 操作返回该患者的全量关联信息，通常包括患者基本信息、就诊记录、检验结果、诊断、用药等相关资源。

该接口适用于患者全景视图、病历汇总、跨资源数据聚合等场景。

本示例展示了调用患者 `$everything` 操作以及结合 `_count` 控制单页返回条数的方式。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>GET</code> 。
URL	String	是	操作地址，格式为 <code>[baseUrl]/Patient/[id]/\$everything</code> 。
Authorization	String	是	访问令牌，格式为 <code>Bearer <AccessToken></code> 。获取方式详见 调用方式 。
Accept	String	否	响应格式，建议使用 <code>application/fhir+json</code> 。
id	String	是	患者资源唯一 ID。可通过创建资源时返回的 <code>id</code> 字段获取，或通过 基础搜索接口 查询患者列表后获取。
_count	Integer	否	控制单页返回记录数。

- ❗ 说明：
- `$everything` 用于返回与目标患者相关联的多种资源。
 - 返回结果通常为 `Bundle`，并可能因数据量较大而分页返回。
 - 可结合 `_count` 参数控制单页结果数量。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，响应体一般为 `Bundle` 资源，包含与指定患者相关的资源集合及分页信息。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 Bundle 。
id	String	Bundle 资源的唯一 ID，通常为 UUID。
meta	Object	Bundle 元数据，包含 lastUpdated 等字段。
meta.lastUpdated	String	Bundle 生成时间，ISO 8601 格式。
type	String	Bundle 类型，通常为 searchset 。
total	Integer	命中的总资源数量。
link	Array	分页链接信息，如 self 、 next 等。
entry	Array	关联资源列表。
entry[].fullUrl	String	资源的完整访问地址。
entry[].resource	Object	与患者相关的资源内容。
entry[].resource.id	String	资源的唯一 ID。
entry[].resource.meta	Object	资源元数据，包含 versionId 、 lastUpdated 等。
entry[].search.mode	String	搜索命中模式，通常为 match 。

示例

请求示例

- 示例一：获取指定患者的全部关联资源。

```
GET /INSTANCE_ID/fhir/Patient/1228032/$everything HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

- 示例二：控制单页返回数量。

```
GET /INSTANCE_ID/fhir/Patient/1228032/$everything?_count=50 HTTP/1.1
Host: HOSTNAME
Authorization: Bearer <AccessToken>
Accept: application/fhir+json
```

响应示例

```
{
  "resourceType": "Bundle",
  "id": "c48fc429-6091-4f61-abce-62388b08ccbc",
  "meta": {
    "lastUpdated": "2026-07-09T18:45:41.430+08:00"
  },
  "type": "searchset",
  "total": 3,
  "link": [
    {
      "relation": "self",
      "url":
"https://HOSTNAME/INSTANCE_ID/fhir/Patient/1228032/$everything"
    }
  ],
  "entry": [
    {
      "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Patient/1228032",
      "resource": {
        "resourceType": "Patient",
        "id": "1228032",
        "meta": {
          "versionId": "1",
          "lastUpdated": "2022-10-10T07:07:33.086-04:00"
        }
      },
      "search": {
        "mode": "match"
      }
    },
    {

```



```
"fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Encounter/enc-20001",
"resource": {
  "resourceType": "Encounter",
  "id": "enc-20001",
  "meta": {
    "versionId": "1",
    "lastUpdated": "2022-10-11T08:30:00.000+08:00"
  },
  "subject": {
    "reference": "Patient/1228032"
  }
},
"search": {
  "mode": "match"
}
},
{
  "fullUrl": "https://HOSTNAME/INSTANCE_ID/fhir/Observation/obs-30001",
  "resource": {
    "resourceType": "Observation",
    "id": "obs-30001",
    "meta": {
      "versionId": "1",
      "lastUpdated": "2022-10-12T09:15:00.000+08:00"
    },
    "subject": {
      "reference": "Patient/1228032"
    }
  },
  "search": {
    "mode": "match"
  }
}
]
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	请求参数格式错误，或患者 ID 非法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无访问该患者全量数据的权限
404 Not Found	指定患者不存在
422 Unprocessable Entity	请求语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

Bundle 请求接口

基础 Bundle 事务

最近更新时间：2026-07-28 16:18:01

接口描述

用于通过单个 `Bundle` 请求提交一个或多个 FHIR 交互操作。客户端将多个交互条目封装在 `Bundle.entry` 中，通过 HTTP `POST` 提交到服务基地址，由服务端按 `Bundle.type` 指定的模式进行处理。

- 在事务模式下，所有条目会作为一个原子事务执行。
- 在批处理模式下，各条目独立执行。

基础事务 `Bundle` 的关键元素包括：

- `Bundle.type`：指定处理模式，如 `transaction` 或 `batch`。
- `Bundle.entry`：交互条目数组，每个条目代表一次独立 REST 交互。
- `Bundle.entry.fullUrl`：条目的完整 URL 或临时 URI。
- `Bundle.entry.resource`：请求体资源内容。
- `Bundle.entry.request`：请求方法、请求 URL 及其他请求控制信息。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>POST</code> 。
URL	String	是	<code>Bundle</code> 提交地址，通常为 <code>[baseUrl]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。
Content-Type	String	否	请求体 MIME 类型，写入类请求通常为 <code>application/fhir+json</code> ；未指定时服务端将根据请求内容自动推断。
Request Body	JSON Object	是	<code>Bundle</code> 资源内容。

请求体主要字段说明：

字段	类型	是否必填	说明
resourceType	String	是	固定为 Bundle 。
type	String	否	处理模式。取值如下： - transaction：事务模式，所有条目作为一个原子事务执行，任一条目失败则整体回滚。 - batch：批处理模式，各条目独立执行，互不影响。 未指定时服务端默认按 transaction 模式处理。
entry	Array	是	交互条目列表。
entry[].fullUrl	String	否	条目的唯一地址或占位 URI。
entry[].resource	Object	否	资源内容，适用于创建、更新、删除、补丁等操作。
entry[].request.method	String	是	条目请求方法，如 POST 、 PUT 、 DELETE 、 PATCH 。
entry[].request.url	String	是	条目对应的目标资源路径。

输出参数

接口调用成功后，通常返回 HTTP 状态码 200 OK 或 201 Created ，并在响应头中包含内容类型及请求追踪信息，响应体一般为 Bundle 资源，用于描述各条目执行结果。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 200 OK 或 201 Created 。
Location	String	Bundle 响应资源地址，格式为 [baseUrl]/Bundle/[bundleId] 。
Content-Location	String	Bundle 响应资源当前版本地址。
Content-Type	String	响应体 MIME 类型，通常为 application/fhir+json 。

x-request-id	String	请求追踪 ID。
--------------	--------	----------

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>Bundle</code> 。
id	String	<code>Bundle</code> 资源的唯一标识。
type	String	响应 <code>Bundle</code> 类型，事务模式返回 <code>transaction-response</code> ，批处理模式返回 <code>batch-response</code> 。
link	Array	链接信息列表，包含与当前 <code>Bundle</code> 相关的导航链接。
link[].relation	String	链接关系类型，如 <code>self</code> （指向当前请求地址）。
link[].url	String	链接地址。
entry	Array	各条目处理结果。
entry[].response.status	String	条目处理状态，如 <code>201 Created</code> 。
entry[].response.location	String	资源位置或版本地址。
entry[].response.etag	String	资源版本标识。
entry[].response.lastModified	String	资源最后修改时间。
entry[].response.outcome	Object	操作结果详情，包含 <code>OperationOutcome</code> 资源，描述条目执行的具体结果信息。

示例

请求示例

```
POST /INSTANCE_ID/fhir HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Bundle",
  "type": "transaction",
  "entry": [
    {
      "fullUrl": "urn:uuid:850bc2ca-d9ab-467b-9924-0e08d0a6e586",
      "resource": {
        "resourceType": "Patient",
        "identifier": [
          {
            "system": "https://example.org/mrns",
            "value": "8fffd03ee-fb56-441d-a3ef-01cdc9f94d89"
          }
        ],
        "name": [
          {
            "use": "official",
            "family": "Trantow673",
            "given": [
              "Matthew562"
            ],
            "prefix": [
              "Mr."
            ]
          }
        ],
        "gender": "male"
      },
      "request": {
        "method": "POST",
        "url": "Patient"
      }
    }
  ]
}
```

```
]
}
```

响应示例

```
HTTP/1.1 200 OK
Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<BUNDLE_ID>
Content-Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<BUNDLE_ID>
Content-Type: application/fhir+json
x-request-id: <REQUEST_ID>
```

```
{
  "resourceType": "Bundle",
  "id": "<BUNDLE_ID>",
  "type": "transaction-response",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir"
    }
  ],
  "entry": [
    {
      "response": {
        "status": "201 Created",
        "location": "Patient/<RESOURCE_ID>/_history/1",
        "etag": "1",
        "lastModified": "2026-07-09T18:50:33.309+08:00",
        "outcome": {
          "resourceType": "OperationOutcome",
          "issue": [
            {
              "severity": "information",
              "code": "informational",
              "details": {
                "coding": [
                  {

```

```
        "system": "https://hapifhir.io/fhir/CodeSystem/hapi-
fhir-storage-response-code",
        "code": "SUCCESSFUL_CREATE",
        "display": "Create succeeded."
      }
    ]
  },
  "diagnostics": "Successfully created resource
\"Patient/<RESOURCE_ID>/_history/1\". Took 68ms."
}
]
}
}
}
]
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	Bundle 结构错误，或条目请求格式不合法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无执行 Bundle 操作的权限
404 Not Found	条目中的目标资源路径不存在
422 Unprocessable Entity	Bundle 语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

捆绑多个关联资源

最近更新时间：2026-07-28 16:18:01

接口描述

用于在同一个 `Bundle` 事务中一次性提交多个彼此关联的资源。该方式适合将患者及其关联的检验、就诊、诊断等资源统一写入，确保这些资源在同一次提交中被整体处理。

在已知资源标识的情况下，可以使用客户端自定义 ID，并在资源之间直接通过这些 ID 建立引用关系。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>POST</code> 。
URL	String	是	<code>Bundle</code> 提交地址，格式为 <code>[baseUrl]</code> 。
Authoriza tion	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。
Content-Type	String	是	请求体 MIME 类型，示例为 <code>application/fhir+json</code> 。
Request Body	JSON Object	是	包含多个关联资源的事务 <code>Bundle</code> 。

请求体主要字段说明：

字段	类型	是否必填	说明
resourceType	String	是	固定为 <code>Bundle</code> 。
type	String	否	<code>Bundle</code> 类型，指定事务处理方式。取值如下： <code>transaction</code>：事务模式，所有条目作为整体提交，任一条目失败则全部回滚。 <code>batch</code>：批处理模式，各条目独立处理，互不影响，详见 FHIR BundleType。

			未指定时服务端默认按 <code>transaction</code> 模式处理。
<code>entry</code>	<code>Array</code>	是	多个资源条目。
<code>entry[].fullUrl</code>	<code>String</code>	否	资源在 Bundle 中的唯一标识。
<code>entry[].resource.id</code>	<code>String</code>	否	客户端指定资源 ID。使用客户端自定义 ID 时，需配合 <code>entry[].request.method</code> 为 <code>PUT</code> 、 <code>entry[].request.url</code> 为 <code>[resourceType]/[id]</code> ，服务端将以此 ID 创建或更新资源；若不指定，由服务端自动生成。
<code>entry[].resource.subject.reference</code>	<code>String</code>	否	对其他资源的引用。
<code>entry[].request.method</code>	<code>String</code>	是	条目请求方法。
<code>entry[].request.url</code>	<code>String</code>	是	条目目标资源路径。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK` 或 `201 Created`，并在响应头中包含内容类型及请求追踪信息，响应体一般为处理结果 Bundle，包含各资源条目的执行状态。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 或 <code>201 Created</code> 。
Content-Type	String	响应体 MIME 类型，通常为 <code>application/fhir+json</code> 。
x-request-id	String	请求追踪 ID。

响应体主要字段说明：

字段	类型	说明
<code>resourceType</code>	String	返回资源类型，通常为 <code>Bundle</code> 。

id	String	Bundle 的唯一标识，由服务端生成。
type	String	响应 Bundle 类型，事务响应为 <code>transaction-response</code> ，批处理响应为 <code>batch-response</code> 。
link	Array	Bundle 关联链接列表。
link[].relation	String	链接关系类型，如 <code>self</code> 表示当前请求自身。
link[].url	String	链接地址。
entry	Array	各资源条目的执行结果。
entry[].response.status	String	条目执行状态。
entry[].response.location	String	资源位置或版本地址。
entry[].response.etag	String	资源版本标识。

示例

请求示例

```
POST /INSTANCE_ID/fhir HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Bundle",
  "type": "transaction",
  "entry": [
    {
      "fullUrl": "Patient/PTA",
      "resource": {
```

```
"resourceType": "Patient",
"id": "PTA",
"identifier": [
  {
    "system": "http://acme.org/mrns",
    "value": "013872"
  }
],
"name": [
  {
    "family": "Simpson",
    "given": [
      "Homer"
    ]
  }
],
"gender": "male"
},
"request": {
  "method": "PUT",
  "url": "Patient/PTA"
}
},
{
  "fullUrl": "Condition/CD1",
  "resource": {
    "resourceType": "Condition",
    "id": "CD1",
    "clinicalStatus": {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-clinical",
          "code": "active"
        }
      ]
    },
    "verificationStatus": {
      "coding": [
        {
```

```

        "system":
"http://terminology.hl7.org/CodeSystem/condition-ver-status",
        "code": "confirmed"
    }
]
},
"category": [
{
    "coding": [
        {
            "system":
"http://terminology.hl7.org/CodeSystem/condition-category",
            "code": "problem-list-item",
            "display": "Problem List Item"
        }
    ]
}
],
"code": {
    "coding": [
        {
            "system": "http://snomed.info/sct",
            "code": "59621000",
            "display": "Essential hypertension"
        }
    ]
},
"subject": {
    "reference": "Patient/PTA"
}
},
"request": {
    "method": "PUT",
    "url": "Condition/CD1"
}
},
{
    "fullUrl": "Condition/CD2",
    "resource": {
        "resourceType": "Condition",

```

```

    "id": "CD2",
    "clinicalStatus": {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-clinical",
          "code": "active"
        }
      ],
    },
    "verificationStatus": {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-ver-status",
          "code": "confirmed"
        }
      ],
    },
    "category": [
      {
        "coding": [
          {
            "system":
"http://terminology.hl7.org/CodeSystem/condition-category",
            "code": "problem-list-item",
            "display": "Problem List Item"
          }
        ]
      }
    ],
    "code": {
      "coding": [
        {
          "system": "http://snomed.info/sct",
          "code": "44054006",
          "display": "Type 2 diabetes mellitus"
        }
      ]
    },
  },

```

```

    "subject": {
      "reference": "Patient/PTA"
    }
  },
  "request": {
    "method": "PUT",
    "url": "Condition/CD2"
  }
}
]
}

```

响应示例

```

HTTP/1.1 200 OK
Content-Type: application/fhir+json
x-request-id: <REQUEST_ID>

```

```

{
  "resourceType": "Bundle",
  "id": "b67e4a2f-3c5d-4e8a-9f01-7a2b3c4d5e6f",
  "type": "transaction-response",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir"
    }
  ],
  "entry": [
    {
      "response": {
        "status": "201 Created",
        "location": "Patient/PTA/_history/1",
        "etag": "1"
      }
    },
    {
      "response": {

```

```

    "status": "201 Created",
    "location": "Condition/CD1/_history/1",
    "etag": "1"
  }
},
{
  "response": {
    "status": "201 Created",
    "location": "Condition/CD2/_history/1",
    "etag": "1"
  }
}
]
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	Bundle 中资源关联关系或请求格式错误
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无执行 Bundle 操作的权限
409 Conflict	资源 ID 或引用关系冲突
422 Unprocessable Entity	Bundle 语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

占位符 ID 与引用

最近更新时间：2026-07-28 16:18:01

接口描述

用于在同一个事务 `Bundle` 中建立尚未分配正式资源 ID 的资源之间的引用关系。当客户端在提交前尚不知道目标资源最终 ID 时，可以为资源分配临时占位符 ID，并在同一 `Bundle` 中通过该占位符完成资源引用。服务端在处理事务时，会将占位符 ID 解析为实际资源 ID，并自动替换相关引用关系。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>POST</code> 。
URL	String	是	<code>Bundle</code> 提交地址，通常为 <code>[baseUrl]</code> 。
Authoriza tion	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。
Content- Type	String	是	请求体 MIME 类型，示例为 <code>application/fhir+json</code> 。
Request Body	JSON Object	是	包含占位符 ID 与引用关系的事务 <code>Bundle</code> 。

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir
```

请求体主要字段说明：

字段	类型	是否必填	说明
<code>entry[].fullUrl</code>	String	否	占位符 ID，通常使用 <code>urn:uuid:{uuid}</code> 形式。当该条目需要被同一 <code>Bundle</code> 内其他条目引用时必须填写；仅创建资源而不被引用时可省略。

entry[].resource	Object	是	对应资源内容。
entry[].resource.subject.reference	String	否	指向其他条目的占位符引用。
entry[].request.method	String	是	条目请求方法，示例为 <code>POST</code> 。
entry[].request.url	String	是	条目目标资源路径。

- ❗ 说明：
- 占位符 ID 应由客户端随机生成。
 - 占位符 ID 仅在当前 Bundle 内部用于建立关联关系。
 - 事务处理完成后，服务端会为新建资源分配正式 ID，并替换引用。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK` 或 `201 Created`，并在响应头中包含内容类型及请求追踪信息，响应体一般为结果 Bundle，包含实际写入后的资源位置和处理状态。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 或 <code>201 Created</code> 。
Content-Type	String	响应体 MIME 类型，通常为 <code>application/fhir+json</code> 。
x-request-id	String	请求追踪 ID。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，固定为 <code>Bundle</code> 。
id	String	响应 Bundle 的唯一 ID，由服务端生成。
type	String	响应 Bundle 类型，固定为 <code>transaction-response</code> 。

link	Array	导航链接，包含指向当前请求地址的 <code>self</code> 链接。
link[].relation	String	链接关系类型，通常为 <code>self</code> 。
link[].url	String	链接对应的 URL。
entry	Array	条目执行结果。
entry[].response.status	String	条目执行状态，如 <code>201 Created</code> 。
entry[].response.location	String	新资源的实际位置。
entry[].response.etag	String	资源版本标识。
entry[].response.lastModified	String	资源最后修改时间。
entry[].response.outcome	Object	条目执行结果详情，为 <code>OperationOutcome</code> 资源，包含执行状态信息。

示例

请求示例

```
POST /INSTANCE_ID/fhir HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Bundle",
  "type": "transaction",
  "entry": [
    {
      "fullUrl": "urn:uuid:e16eac01-a5ee-4904-b1c8-f4bd56e338d5",
      "resource": {
```

```
"resourceType": "Patient",
"identifier": [
  {
    "system": "http://acme.org/mrns",
    "value": "013872"
  }
],
"name": [
  {
    "family": "Simpson",
    "given": [
      "Homer"
    ]
  }
],
"gender": "male"
},
"request": {
  "method": "POST",
  "url": "Patient"
}
},
{
  "fullUrl": "urn:uuid:499733fe-7ced-4d15-81ce-8a433a1fb71e",
  "resource": {
    "resourceType": "Condition",
    "clinicalStatus": {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-clinical",
          "code": "active"
        }
      ]
    },
    "verificationStatus": {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-ver-status",
```

```

        "code": "confirmed"
      }
    ]
  },
  "category": [
    {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-category",
          "code": "problem-list-item",
          "display": "Problem List Item"
        }
      ]
    }
  ],
  "code": {
    "coding": [
      {
        "system": "http://snomed.info/sct",
        "code": "59621000",
        "display": "Essential hypertension"
      }
    ]
  },
  "subject": {
    "reference": "urn:uuid:e16eac01-a5ee-4904-b1c8-f4bd56e338d5"
  }
},
"request": {
  "method": "POST",
  "url": "Condition"
}
}
]
}

```

响应示例

```
HTTP/1.1 200 OK
Content-Type: application/fhir+json;charset=UTF-8
Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/bundle-response-id
Content-Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/bundle-
response-id
x-request-id: <REQUEST_ID>
```

```
{
  "resourceType": "Bundle",
  "id": "bundle-response-id",
  "type": "transaction-response",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir"
    }
  ],
  "entry": [
    {
      "response": {
        "status": "201 Created",
        "location": "Patient/patient-resource-id/_history/1",
        "etag": "1",
        "lastModified": "2022-02-23T10:00:00.000+08:00",
        "outcome": {
          "resourceType": "OperationOutcome",
          "issue": [
            {
              "severity": "information",
              "code": "informational",
              "details": {
                "coding": [
                  {
                    "system": "https://hapifhir.io/fhir/CodeSystem/hapi-
fhir-storage-response-code",
                    "code": "SUCCESSFUL_CREATE",
                    "display": "Create succeeded."
                  }
                ]
              }
            }
          ]
        }
      }
    }
  ]
}
```

```

        ]
      },
      "diagnostics": "Successfully created resource
\"Patient/patient-resource-id/_history/1\"."
    }
  ]
}
},
{
  "response": {
    "status": "201 Created",
    "location": "Condition/condition-resource-id/_history/1",
    "etag": "1",
    "lastModified": "2022-02-23T10:00:00.000+08:00",
    "outcome": {
      "resourceType": "OperationOutcome",
      "issue": [
        {
          "severity": "information",
          "code": "informational",
          "details": {
            "coding": [
              {
                "system": "https://hapifhir.io/fhir/CodeSystem/hapi-
fhir-storage-response-code",
                "code": "SUCCESSFUL_CREATE",
                "display": "Create succeeded."
              }
            ]
          },
          "diagnostics": "Successfully created resource
\"Condition/condition-resource-id/_history/1\"."
        }
      ]
    }
  }
}
}
]

```

```
}

```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	占位符 ID 格式错误，或引用路径不合法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无执行 Bundle 操作的权限
409 Conflict	引用关系冲突或资源状态冲突
422 Unprocessable Entity	Bundle 语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

条件创建

最近更新时间：2026-07-28 16:18:01

接口描述

用于在事务 Bundle 中执行条件创建操作。客户端在提交待创建资源的同时，会提供一段搜索条件，服务端会先按该条件查询是否已存在匹配资源，若不存在则创建，若已存在则不再重复创建。该机制适合避免重复写入患者、检验结果等具有自然唯一标识的数据。

在事务 Bundle 中，条件创建通过 Bundle.entry.request.ifNotExist 指定，请求方法仍为 POST 。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 POST 。
URL	String	是	Bundle 提交地址，通常为 [baseUrl] 。
Authorization	String	是	身份认证令牌，格式为 Bearer <AccessToken> ，AccessToken 通过 调用方式 中的 GetAccessToken 接口获取（实例控制台场景可通过 GetWebAccessToken 接口获取）。
Content-Type	String	是	请求体 MIME 类型，示例为 application/fhir+json 。
Request Body	JSON Object	是	包含条件创建条目的事务 Bundle。

请求体主要字段说明：

字段	类型	是否必填	说明
entry[].resource	Object	是	待创建资源内容。
entry[].request.method	String	是	固定为 POST 。
entry[].request.url	String	是	目标资源类型路径。

entry[].request.ifNotExist	String	否	条件创建搜索表达式。省略时该条目退化为普通创建（POST），不进行条件匹配。
----------------------------	--------	---	--

- ❗ 说明：
- 若 ifNotExist 查询无匹配资源，则执行创建。
 - 若查询已有匹配资源，则不重复创建。
 - 条件表达式通常使用业务唯一标识字段，如 identifier。

输出参数

接口调用成功后，通常返回 HTTP 状态码 200 OK 或 201 Created，并在响应头中包含内容类型及请求追踪信息，响应体一般为结果 Bundle，包含各条目的执行结果及资源位置。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 200 OK 或 201 Created。
Content-Type	String	响应体 MIME 类型，通常为 application/fhir+json。
x-request-id	String	请求追踪 ID。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 Bundle。
id	String	结果 Bundle 的唯一标识，由服务端分配。
type	String	结果 Bundle 类型，事务响应固定为 transaction-response。
link	Array	分页相关链接信息，包含 relation 与 url 字段。
entry	Array	条目执行结果。

entry[].response.status	String	条目执行状态。
entry[].response.location	String	新建或匹配资源的位置。
entry[].response.etag	String	条目对应资源的版本标识。

示例

请求示例

```
POST /INSTANCE_ID/fhir HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Bundle",
  "type": "transaction",
  "entry": [
    {
      "fullUrl": "urn:uuid:95dbbf93-5829-46ba-9021-2545a1da3aa5",
      "resource": {
        "resourceType": "Patient",
        "identifier": [
          {
            "system": "http://acme.org/mrns",
            "value": "013873"
          }
        ],
        "name": [
          {
            "family": "Simpson",
            "given": [
              "Homer"
            ]
          }
        ]
      }
    ]
  ]
}
```

```
    "gender": "male"
  },
  "request": {
    "method": "POST",
    "url": "Patient",
    "ifNotExist": "Patient?identifier=http://acme.org/mrns|013873"
  }
},
{
  "fullUrl": "urn:uuid:124ff3c8-f251-4bd9-8c44-cc6568180eae",
  "resource": {
    "resourceType": "Condition",
    "identifier": [
      {
        "system": "http://acme.org/cond",
        "value": "46253"
      }
    ],
    "clinicalStatus": {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-clinical",
          "code": "active"
        }
      ]
    },
    "verificationStatus": {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-ver-status",
          "code": "confirmed"
        }
      ]
    }
  },
  "category": [
    {
      "coding": [
        {
```

```
      "system":
"http://terminology.hl7.org/CodeSystem/condition-category",
      "code": "problem-list-item",
      "display": "Problem List Item"
    }
  ]
}
],
"code": {
  "coding": [
    {
      "system": "http://snomed.info/sct",
      "code": "59621000",
      "display": "Essential hypertension"
    }
  ]
},
"subject": {
  "reference": "urn:uuid:95dbbf93-5829-46ba-9021-2545a1da3aa5"
}
},
"request": {
  "method": "POST",
  "url": "Condition",
  "ifNotExist": "Condition?identifier=http://acme.org/cond|46253"
}
}
]
```

响应示例

```
HTTP/1.1 200 OK
Content-Type: application/fhir+json
x-request-id: <REQUEST_ID>
```

```
{
  "resourceType": "Bundle",
```

```

    "id": "d139061a-6078-4670-85f7-f4598e571ec7",
    "type": "transaction-response",
    "link": [
      {
        "relation": "self",
        "url": "https://HOSTNAME/INSTANCE_ID/fhir"
      }
    ],
    "entry": [
      {
        "response": {
          "status": "201 Created",
          "location": "https://HOSTNAME/INSTANCE_ID/fhir/Patient/95dbbf93-5829-46ba-9021-2545a1da3aa5/_history/1",
          "etag": "W/\"1\""
        }
      },
      {
        "response": {
          "status": "201 Created",
          "location": "https://HOSTNAME/INSTANCE_ID/fhir/Condition/124ff3c8-f251-4bd9-8c44-cc6568180eae/_history/1",
          "etag": "W/\"1\""
        }
      }
    ]
  }

```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	条件创建表达式格式错误
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无执行 Bundle 操作的权限

409 Conflict	匹配结果不唯一或资源状态冲突
412 Precondition Failed	条件创建搜索表达式匹配到多条资源，无法确定唯一目标
422 Unprocessable Entity	Bundle 语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

条件更新

最近更新时间：2026-07-28 16:18:02

接口描述

用于在事务 `Bundle` 中执行条件更新操作。客户端提交资源内容时，不直接指定目标资源 ID，而是提供一段搜索条件；服务端先根据该条件查询目标资源，再决定执行更新或创建。

条件更新的处理规则通常如下：

- 若无匹配资源，则创建新资源
- 若存在匹配资源，则使用 `Bundle.entry.resource` 中的内容更新该资源

在事务 `Bundle` 中，请求方法通常为 `PUT`，搜索表达式放在 `Bundle.entry.request.url` 中。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>POST</code> 。
URL	String	是	<code>Bundle</code> 提交地址，格式为 <code>[baseUrl]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。
Content-Type	String	否	请求体 MIME 类型，写入类请求通常为 <code>application/fhir+json</code> ；未指定时服务端将根据请求内容自动推断。
Request Body	JSON Object	是	包含条件更新条目的事务 <code>Bundle</code> 。

请求 URL 示例：

```
https://HOSTNAME/INSTANCE_ID/fhir
```

请求体主要字段说明：

字段	类型	是否必填	说明
----	----	------	----

entry[].resource	Object	是	待更新资源内容
entry[].request.method	String	是	固定为 <code>PUT</code>
entry[].request.url	String	是	条件更新搜索表达式

- ❗ 说明：
- 条件更新不直接写死资源 ID，而是通过搜索条件定位目标资源。
 - 若匹配结果为空，则通常按创建处理。
 - 若匹配结果唯一，则对该资源执行更新。

输出参数

接口调用成功后，返回 HTTP 状态码 `200 OK`，并在响应头中包含内容类型及请求追踪信息，响应体为结果 Bundle，包含各条目的执行状态与资源位置。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 。
Location	String	结果 Bundle 的访问地址。
Content-Location	String	结果 Bundle 的内容定位地址。
Content-Type	String	响应体 MIME 类型，通常为 <code>application/fhir+json</code> 。
x-request-id	String	请求追踪 ID。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>Bundle</code> 。
id	String	结果 Bundle 的唯一 ID。
type	String	Bundle 类型，条件更新返回 <code>transaction-response</code> 。

link	Array	导航链接，包含 <code>relation</code> 与 <code>url</code> 字段， <code>relation</code> 为 <code>self</code> 时指向当前请求地址。
entry	Array	条目执行结果。
entry[].response.status	String	条目执行状态。匹配到已有资源时返回 <code>200 OK</code> ；无匹配资源导致新建时返回 <code>201 Created</code> 。
entry[].response.location	String	更新后或新建资源的位置。
entry[].response.etag	String	资源版本标识。
entry[].response.outcome	Object	条目执行详情，包含 <code>OperationOutcome</code> 信息。

示例

请求示例

```
POST /INSTANCE_ID/fhir HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Bundle",
  "type": "transaction",
  "entry": [
    {
      "fullUrl": "urn:uuid:95dbbf93-5829-46ba-9021-2545a1da3aa5",
      "resource": {
        "resourceType": "Patient",
        "identifier": [
          {
            "system": "http://acme.org/mrns",
            "value": "013873"
          }
        ],
        "name": [
```

```

        {
          "family": "Simpson",
          "given": [
            "Homer"
          ]
        }
      ],
      "gender": "male"
    },
    "request": {
      "method": "PUT",
      "url": "Patient?identifier=http://acme.org/mrns|013873"
    }
  },
  {
    "fullUrl": "urn:uuid:124ff3c8-f251-4bd9-8c44-cc6568180eae",
    "resource": {
      "resourceType": "Condition",
      "identifier": [
        {
          "system": "http://acme.org/cond",
          "value": "46253"
        }
      ],
      "clinicalStatus": {
        "coding": [
          {
            "system":
"http://terminology.hl7.org/CodeSystem/condition-clinical",
            "code": "active"
          }
        ]
      },
      "verificationStatus": {
        "coding": [
          {
            "system":
"http://terminology.hl7.org/CodeSystem/condition-ver-status",
            "code": "confirmed"
          }
        ]
      }
    }
  }

```

```

    ]
  },
  "category": [
    {
      "coding": [
        {
          "system":
"http://terminology.hl7.org/CodeSystem/condition-category",
          "code": "problem-list-item",
          "display": "Problem List Item"
        }
      ]
    }
  ],
  "code": {
    "coding": [
      {
        "system": "http://snomed.info/sct",
        "code": "59621000",
        "display": "Essential hypertension"
      }
    ]
  },
  "subject": {
    "reference": "urn:uuid:95dbbf93-5829-46ba-9021-2545a1da3aa5"
  }
},
"request": {
  "method": "PUT",
  "url": "Condition?identifier=http://acme.org/cond|46253"
}
}
]
}

```

响应示例

```

HTTP/1.1 200 OK
Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<bundle-id>

```

```
Content-Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<bundle-id>
Content-Type: application/fhir+json
x-request-id: <REQUEST_ID>
```

```
{
  "resourceType": "Bundle",
  "id": "<bundle-id>",
  "type": "transaction-response",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir"
    }
  ],
  "entry": [
    {
      "response": {
        "status": "200 OK",
        "location": "Patient/95dbbf93-5829-46ba-9021-2545a1da3aa5/_history/2",
        "etag": "2"
      }
    },
    {
      "response": {
        "status": "200 OK",
        "location": "Condition/124ff3c8-f251-4bd9-8c44-cc6568180eae/_history/2",
        "etag": "2"
      }
    }
  ]
}
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
-----	----

400 Bad Request	条件更新表达式格式错误
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无执行 Bundle 操作的权限
409 Conflict	资源状态冲突
412 Precondition Failed	条件更新搜索条件匹配到多个资源，无法确定唯一更新目标
422 Unprocessable Entity	Bundle 语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

事务内删除

最近更新时间：2026-07-28 16:18:02

接口描述

用于在事务 `Bundle` 中执行删除操作。客户端可在同一个事务请求中提交多个删除条目，由服务端统一处理。在事务模式下，删除操作以整体方式执行，即如果其中任意一个删除条目失败，则整个事务回滚，其他删除也不会生效。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>POST</code> 。
URL	String	是	<code>Bundle</code> 提交地址，通常为 <code>[baseUrl]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， <code>AccessToken</code> 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。
Content-Type	String	否	请求体 MIME 类型，写入类请求通常为 <code>application/fhir+json</code> ；未指定时服务端将根据请求内容自动推断。
Request Body	JSON Object	是	包含删除条目的事务 <code>Bundle</code> 。

请求体主要字段说明：

字段	类型	是否必填	说明
<code>resourceType</code>	String	是	固定为 <code>Bundle</code> 。
<code>type</code>	String	否	处理模式。取值为 <code>transaction</code> （事务模式，所有条目作为一个原子事务执行，任一条目失败则整体回滚）或 <code>batch</code> （批处理模式，各条目独立执行，互不影响）。未指定时默认按 <code>transaction</code> 处理。
<code>entry</code>	Array	是	删除条目列表。

entry[].request.method	String	是	固定为 <code>DELETE</code> 。
entry[].request.url	String	是	待删除资源路径，格式为 <code>[resourceType]/[id]</code> ，例如 <code>Patient/A0</code> 。

输出参数

接口调用成功后，返回 HTTP 状态码 `200 OK` ，并在响应头中包含内容类型及请求追踪信息，响应体为 `transaction-response` 类型的结果 Bundle，包含各删除条目的执行状态。需要注意的是，每个删除条目的响应状态码为 `204 No Content` ，而非整体响应的 `200 OK` 。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 。
Location	String	Bundle 响应资源地址，格式为 <code>[baseUrl]/Bundle/[bundleId]</code> 。
Content-Location	String	Bundle 响应资源当前版本地址。
Content-Type	String	响应体 MIME 类型，通常为 <code>application/fhir+json</code> 。
x-request-id	String	请求追踪 ID。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>Bundle</code> 。
id	String	Bundle 资源的唯一标识。
type	String	响应 Bundle 类型，事务模式返回 <code>transaction-response</code> ，批处理模式返回 <code>batch-response</code> 。
link	Array	链接信息列表，包含与当前 Bundle 相关的导航链接。

link[].relation	String	链接关系类型，如 <code>self</code> （指向当前请求地址）。
link[].url	String	链接地址。
entry	Array	各删除条目的执行结果。
entry[].response.status	String	删除执行状态，成功时为 <code>204 No Content</code> 。
entry[].response.location	String	删除后资源版本位置。
entry[].response.etag	String	资源版本标识。
entry[].response.outcome	Object	操作结果详情，包含 <code>OperationOutcome</code> 资源，描述条目执行的具体结果信息。

示例

请求示例

```
POST /INSTANCE_ID/fhir HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Bundle",
  "type": "transaction",
  "entry": [
    {
      "request": {
        "method": "DELETE",
        "url": "Patient/A0"
      }
    },
    {
      "request": {
```

```
    "method": "DELETE",
    "url": "Patient/A1"
  }
}
]
```

响应示例

```
HTTP/1.1 200 OK
Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<BUNDLE_ID>
Content-Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<BUNDLE_ID>
Content-Type: application/fhir+json
x-request-id: <REQUEST_ID>
```

```
{
  "resourceType": "Bundle",
  "id": "<BUNDLE_ID>",
  "type": "transaction-response",
  "link": [
    {
      "relation": "self",
      "url": "https://HOSTNAME/INSTANCE_ID/fhir"
    }
  ],
  "entry": [
    {
      "response": {
        "status": "204 No Content",
        "location": "Patient/A0/_history/2",
        "etag": "2",
        "outcome": {
          "resourceType": "OperationOutcome",
          "issue": [
            {
              "severity": "information",
              "code": "informational",
              "details": {
```

```

        "coding": [
            {
                "system": "https://hapifhir.io/fhir/CodeSystem/hapi-
fhir-storage-response-code",
                "code": "SUCCESSFUL_DELETE",
                "display": "Delete succeeded."
            }
        ]
    },
    "diagnostics": "Successfully deleted 1 resource(s). Took
87ms."
}
]
}
}
},
{
    "response": {
        "status": "204 No Content",
        "location": "Patient/A1/_history/2",
        "etag": "2",
        "outcome": {
            "resourceType": "OperationOutcome",
            "issue": [
                {
                    "severity": "information",
                    "code": "informational",
                    "details": {
                        "coding": [
                            {
                                "system": "https://hapifhir.io/fhir/CodeSystem/hapi-
fhir-storage-response-code",
                                "code": "SUCCESSFUL_DELETE",
                                "display": "Delete succeeded."
                            }
                        ]
                    }
                }
            ],
            "diagnostics": "Successfully deleted 1 resource(s). Took
87ms."
        }
    }
}

```

```
        ]
      }
    }
  }
}
]
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	删除条目格式错误，或资源路径非法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无执行删除操作的权限
404 Not Found	条目中的目标资源不存在
409 Conflict	某一删除条目失败导致事务回滚
500 Internal Server Error	服务端内部处理异常

事务内 Patch

最近更新时间：2026-07-28 16:18:02

接口描述

用于在事务 `Bundle` 中执行补丁更新操作。客户端可将 FHIR Patch 文档作为 Bundle 条目的一部分提交，由服务端在事务执行过程中对指定资源进行局部更新。

当使用 FHIR Patch 时，补丁文档通常放置在 `Bundle.entry.resource` 中。若使用 JSON Patch，则通常需要将补丁内容封装在 `Binary` 资源后再放入条目资源中。条目请求方法固定为 `PATCH`。

输入参数

参数名称	类型	是否必填	说明
HTTP Method	String	是	固定为 <code>POST</code> 。
URL	String	是	Bundle 提交地址，通常为 <code>[baseUrl]</code> 。
Authorization	String	是	身份认证令牌，格式为 <code>Bearer <AccessToken></code> ， AccessToken 通过 调用方式 中的 <code>GetAccessToken</code> 接口获取（实例控制台场景可通过 <code>GetWebAccessToken</code> 接口获取）。
Content-Type	String	否	请求体 MIME 类型，写入类请求通常为 <code>application/fhir+json</code> ；未指定时服务端将根据请求内容自动推断。
Request Body	JSON Object	是	包含 Patch 条目的事务 Bundle。

请求体主要字段说明：

字段	类型	是否必填	说明
resourceType	String	是	固定为 <code>Bundle</code> 。
type	String	否	处理模式。取值为 <code>transaction</code> （事务模式，所有条目作为一个原子事务执行，任一条目失败则整体回滚）或 <code>batch</code> （批处理模式，各条目独立执行，互不影响）。

entry	Array	是	Patch 条目列表。
entry[].resource	Object	是	FHIR Patch 文档内容。使用 FHIR Patch 时为 <code>Parameters</code> 资源，包含补丁操作类型、路径和值；使用 JSON Patch 时需将补丁内容封装在 <code>Binary</code> 资源中。
entry[].request.method	String	是	固定为 <code>PATCH</code> 。
entry[].request.url	String	是	需要执行补丁更新的资源路径，格式为 <code>[resourceType]/[id]</code> ，例如 <code>Patient/123</code> 。

输出参数

接口调用成功后，通常返回 HTTP 状态码 `200 OK`，并在响应头中包含内容类型及请求追踪信息，响应体一般为结果 Bundle，包含各条目执行状态及资源版本信息。

响应头示例说明：

参数名称	类型	说明
Status Code	Integer	成功时返回 <code>200 OK</code> 。
Location	String	Bundle 响应资源地址，格式为 <code>[baseUrl]/Bundle/[bundleId]</code> 。
Content-Location	String	Bundle 响应资源当前版本地址。
Content-Type	String	响应体 MIME 类型，通常为 <code>application/fhir+json</code> 。
x-request-id	String	请求追踪 ID。

响应体主要字段说明：

字段	类型	说明
resourceType	String	返回资源类型，通常为 <code>Bundle</code> 。
id	String	Bundle 资源的唯一标识。

type	String	响应 Bundle 类型，事务模式返回 <code>transaction-response</code> ，批处理模式返回 <code>batch-response</code> 。
link	Array	链接信息列表，包含与当前 Bundle 相关的导航链接。
link[].relation	String	链接关系类型，如 <code>self</code> （指向当前请求地址）。
link[].url	String	链接地址。
entry	Array	各 Patch 条目的执行结果。
entry[].response.status	String	Patch 执行状态。
entry[].response.location	String	更新后资源位置。
entry[].response.etag	String	资源版本标识。
entry[].response.outcome	Object	操作结果详情，包含 <code>OperationOutcome</code> 资源，描述条目执行的具体结果信息。

示例

请求示例

```
POST /INSTANCE_ID/fhir HTTP/1.1
Host: HOSTNAME
Authorization: Bearer YOUR_ACCESS_TOKEN
Content-Type: application/fhir+json
```

```
{
  "resourceType": "Bundle",
  "type": "transaction",
  "entry": [
    {
      "resource": {
        "resourceType": "Parameters",
```

```
"parameter": [
  {
    "name": "operation",
    "part": [
      { "name": "type", "valueCode": "replace" },
      { "name": "path", "valueString": "Patient.gender" },
      { "name": "value", "valueCode": "male" }
    ]
  },
  {
    "name": "operation",
    "part": [
      { "name": "type", "valueCode": "replace" },
      { "name": "path", "valueString": "Patient.identifier" },
      { "name": "value", "valueIdentifier": { "system":
"http://new-system", "value": "0001" } }
    ]
  }
],
"request": { "method": "PATCH", "url": "Patient/___PID___" }
}
```

响应示例

```
HTTP/1.1 200 OK
Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<BUNDLE_ID>
Content-Location: https://HOSTNAME/INSTANCE_ID/fhir/Bundle/<BUNDLE_ID>
Content-Type: application/fhir+json
x-request-id: <REQUEST_ID>
```

```
{
  "resourceType": "Bundle",
  "id": "<BUNDLE_ID>",
  "type": "transaction-response",
  "link": [
```



```
{
  "relation": "self",
  "url": "https://HOSTNAME/INSTANCE_ID/fhir"
},
"entry": [
  {
    "response": {
      "status": "200 OK",
      "location": "Patient/123/_history/2",
      "etag": "2",
      "outcome": {
        "resourceType": "OperationOutcome",
        "issue": [
          {
            "severity": "information",
            "code": "informational",
            "details": {
              "coding": [
                {
                  "system": "https://hapifhir.io/fhir/CodeSystem/hapi-
fhir-storage-response-code",
                  "code": "SUCCESSFUL_PATCH",
                  "display": "Patch succeeded."
                }
              ]
            },
            "diagnostics": "Successfully patched resource
\"Patient/123/_history/2\"."
          }
        ]
      }
    }
  }
]
```

错误码

常见错误码如下，更多错误码请参见 [错误码](#)。

错误码	说明
400 Bad Request	Patch 文档格式错误，或补丁路径不合法
401 Unauthorized	未认证，缺少有效身份凭证
403 Forbidden	已认证但无执行 Patch 操作的权限
404 Not Found	条目中的目标资源不存在
422 Unprocessable Entity	Patch 语法正确，但未通过业务或规则校验
500 Internal Server Error	服务端内部处理异常

错误码

最近更新时间：2026-07-28 16:18:02

本文将为您介绍 THDS 在 FHIR 资源操作、搜索操作以及 Bundle 事务/批处理操作中可能返回的通用错误码。实际接口返回的错误信息可能会因资源类型、请求参数、业务规则和权限配置不同而有所差异，但总体遵循以下约定。

通用错误码说明

错误码	名称	说明	常见触发场景
400	Bad Request	请求格式错误，或请求参数不合法	请求体 JSON 结构错误、资源 ID 非法、搜索参数格式错误、补丁路径错误、Bundle 结构错误
401	Unauthorized	未认证，缺少有效身份凭证	未传认证信息、认证令牌无效、认证令牌已过期
403	Forbidden	已认证但无权访问目标资源或操作	无创建、更新、删除、检索或事务执行权限
404	Not Found	请求的资源、资源类型或路径不存在	指定资源不存在、目标接口路径错误、事务条目中的目标资源不存在
409	Conflict	当前资源状态或匹配结果与操作要求冲突	版本冲突、条件创建匹配不唯一、条件更新匹配不唯一、事务中条目冲突
410	Gone	资源或指定历史版本已不可访问	资源已逻辑删除、历史版本不可访问
415	Unsupported Media Type	请求内容类型不受支持	未使用 <code>application/fhir+json</code> 、未使用 <code>application/json-patch+json</code>
422	Unprocessable Entity	请求语法正确，但未通过业务或规则校验	FHIR 资源结构不完整、业务约束不满足、搜索表达式可解析但不可执行
500	Internal Server Error	服务端内部处理异常	服务执行异常、事务处理异常、未知内部错误

按接口类型说明

资源基础操作

适用于 `Create`、`Read`、`vRead`、`Update`、`Delete`、`Patch` 等资源级接口。常见情况如下：

- **400 Bad Request**：创建或更新时提交的资源内容不符合 JSON/FHIR 结构要求；读取、删除时资源 ID 非法；Patch 时 `op`、`path` 或补丁文档格式错误。
- **403 Forbidden**：当前调用方没有对应资源的创建、读取、更新、删除权限。
- **404 Not Found**：指定资源不存在，或请求的资源类型端点不存在。
- **409 Conflict**：更新时发生版本冲突，删除时当前资源状态不允许删除，Patch 时目标路径与资源当前状态不匹配。
- **410 Gone**：读取的资源或历史版本已不可访问。
- **415 Unsupported Media Type**：创建、更新未使用 `application/fhir+json`，或 Patch 未使用 `application/json-patch+json`。
- **422 Unprocessable Entity**：资源内容语法正确，但未通过业务规则或 FHIR 规则校验。

搜索操作

适用于基础搜索、引用搜索、数值搜索、日期时间搜索、分页、排序、全文搜索以及患者全量信息检索等接口。常见情况如下：

- **400 Bad Request**：搜索参数格式错误，或参数值不合法，例如日期比较前缀错误、数值表达式错误、排序字段错误、分页参数错误。
- **403 Forbidden**：当前调用方没有搜索对应资源或访问患者全量信息的权限。
- **404 Not Found**：指定资源类型不存在，或目标患者不存在。
- **422 Unprocessable Entity**：搜索语法正确，但未通过业务规则、权限规则或资源级约束校验。

Bundle 事务与批处理操作

适用于基础 Bundle 事务、捆绑多个关联资源、占位符 ID 与引用、条件创建、条件更新、事务内删除、事务内 Patch 等接口。常见情况如下：

- **400 Bad Request**：Bundle 结构错误、条目请求格式错误、占位符 ID 非法、引用路径不合法、条件表达式格式错误。
- **404 Not Found**：Bundle 条目中的目标资源路径不存在，或事务操作目标资源不存在。
- **409 Conflict**：资源 ID 冲突、引用关系冲突、条件创建/条件更新匹配不唯一，或事务中某一删除条目失败导致整体回滚。
- **422 Unprocessable Entity**：Bundle 语法正确，但资源内容、引用关系或业务规则校验失败。

使用建议

- 在调用写入类接口前，确认 `Content-Type` 与请求体结构符合要求。
- 在执行更新、删除、Patch 和事务操作前，确认目标资源存在且当前调用方具备操作权限。
- 在使用搜索、条件创建、条件更新等接口时，尽量保证搜索条件具备明确性和唯一性。
- 在处理 409、422 等错误时，建议结合具体响应内容进一步定位字段级或业务级问题。