**DevTest** [**REST APIs**](https://support.itko.com/confluence/display/DEV/REST+APIs)

* [Page restrictions apply](https://support.itko.com/confluence/display/DEV/REST+APIs)
* Added by [Robert M. Williams](https://support.itko.com/confluence/display/~rwilliams), last edited by [Daniel Bingham](https://support.itko.com/confluence/display/~dbingham) on Aug 09, 2017  ([view change](https://support.itko.com/confluence/pages/diffpages.action?pageId=32048790&originalId=37653341))



There has been a fair amount of confusion around the various sets of REST APIs in the product.  This page is intended to clear up that confusion.  Below you will find brief explanations of the various API projects and where you can find documentation about it (if applicable).  It will also be noted which are public and which are not.

**lisa-invoke**

This project dates back many years (2008).  It was the first API published for DevTest (then called "LISA").  This is still shipped with the product.

**Public:** YES

**Documentation:**<https://docops.ca.com/display/DTS101/LISA+Invoke> (and the pages below this)<http://localhost:1505/lisa-invoke/> (very brief – URL accessible when running Registry locally)

**lisa-invoke2**

This also dates back several years (2013) and was the first attempt to replace lisa-invoke with something better.  This is still shipped with the product and is the primary "public" API that customers use today.

**Public:** YES

**Documentation:**  
<https://docops.ca.com/devtest-solutions/10-1/en/reference/rest-invoke-api>  
<http://localhost:1505/api/swagger/> (Old Swagger UI, accessible when running Registry locally)

**lisa-invoke2-pathfinder**

This is an additional project under the "invoke2" umbrella, but I was unable to find any information about it, documentation for it, or any usages of it.  The code is there, but as of this writing, I'm not sure what it is used for.

**Invoke 3**

This is the latest generation of APIs and encapsulates several projects.  I will list them them each separately with their status.

**test-invoke**

This is a collection of REST APIs specific to the "test" side of the product.  These APIs deal with creating tests, deploying them, monitoring them, reporting on them, etc.  As of this writing, *everything* in this project is purely intended as internal support for Phoenix and is not publicly documented or supported.

**Public:** NO

**Documentation:**  
No public docs.  
Developer docs in the LISA repo at: test-invoke/grails-app/controllers/com/ca/test/invoke/UrlMappings.groovy

**pathfinder-invoke**

This is a collection of REST APIs specific to CAI functionality.

**Public:** Yes - Theoretically this is in our docs, but I couldn't get the instructions to work for browsing the API and it probably should NOT be public anyway, but that ship seems to have sailed.

**Documentation:**  
<https://docops.ca.com/devtest-solutions/10-1/en/agents/devtest-java-agent/java-agent-rest-apis>  
Developer docs in the LISA repo at: lisa-remote/pathfinder-invoke/grails-app/controllers/pathfinder/invoke/UrlMappings.groovy

**virtualize-invoke**

This is a collection of REST APIs specific to the "SV" side of the product.  These APIs deal with VSE servers, creating virtual services, deploying them, etc.  As of this writing, there is a mix of public and non-public endpoints in this project.  Endpoints with "v1" in the URL are *not* public.  Endpoints with "v2" in the URL *are* public.

**Public:** Mix – "v1" Endpoints are not, "v2" Endpoints are

Documentation:  
<https://docops.ca.com/devtest-solutions/10-1/en/using/using-the-virtual-services-invoke-api>  
<http://localhost:1505/lisa-virtualize-invoke/api/v2/swagger-ui> (Swagger UI, accessible when running Registry locally)  
<http://localhost:1505/lisa-virtualize-invoke/api/v2/swagger-json> (Swagger 2.0 Spec, accessible when running Registry locally)  
Developer docs in the LISA repo at: virtualize-invoke/grails-app/controllers/lisa/virtualize/invoke/UrlMappings.groovy

**res-hub**

This is a collection of REST APIs specifically for artifact management (i.e. projects, virtual services, tests, configs, etc on disk).  While we could make some of these APIs public with minimal effort, we have not done so yet.

**Public:** NO

**Documentation:**  
No public docs.  
Developer docs in the LISA repo at: res-hub/grails-app/controllers/com/ca/devtest/reshub/UrlMappings.groovy

**Solrservice**

This is a collection of REST APIs specific to searching the CAI database.  These APIs support Phoenix and are intended as internal support and are not publicly documented or supported.

**Public:** NO

**Documentation:**  
No public docs.  
Developer docs in the LISA repo at: lisa-remote/solrservice/grails-app/controllers/com/ca/pf/invoke/UrlMappings.groovy

**Phoenix APIs**

These aren't really APIs in the same sense as the others.  These are not and will never be "public" APIs intended for use by customers, or even other parts of our own product.  These APIs only exist to allow the client side (javascript) and server side (groovy) portions of Phoenix to interact with each other for the purpose of making the app work.

If anyone in the field or at a customer site says that he found an API using browser based developer tools on Phoenix, what he found is one of these APIs.  They should NOT use these APIs directly and should NOT depend on them being stable across releases.  All of these APIs are subject to change without notice and may have side effects or behavior that is unexpected and/or undesirable in the context of direct usage outside of Phoenix.  There is no reason to ever interact with these APIs directly unless you are a developer writing JavaScript for Phoenix.

**Public:** NO

**lisa- virtualize-invoke/api/v1/ Developer Documentation**

**Virtual Service Processor**

A Virtual Service Processor can be created using a model archive.

It can be used to

a) create a request object from raw request.

b) find matching transactions for a raw request

c) Import transactions using request/response pairs

A Virtual Service Processor expires in vs.processor.expiry.minutes (default 10). This property can be configured in  lisa-virtualize-invoke-config.properties or lisa-virtualize-invoke-config.groovy

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Verb** | **URL** | **Purpose** | **Accepts** | **Returns** | **Product Area** |
| POST | /lisa-virtualize-invoke/api/v1/vse/vsProcessor | Create Virtual Service Processor. | Model Archive bytes in request body | application/json  {  \_fields: [ ]  expiryTime: "2014-09-267 18:38:00 UTC"  findAllMatchingTransactionsLocation: /lisa-virtualize-invoke/api/v1/vse/vsProcessor/b9e24713-eb8a-48cf-a42f-478ff61ee56b/findAllMatchingTransactions  id: "b9e24713-eb8a-48cf-a42f-478ff61ee56b"  importRequestResponsePairsLocation: /lisa-virtualize-invoke/api/v1/vse/vsProcessor/b9e24713-eb8a-48cf-a42f-478ff61ee56b/importRequestResponsePairs  location: /lisa-virtualize-invoke/api/v1/vse/vsProcessor/b9e24713-eb8a-48cf-a42f-478ff61ee56b  showRequestLocation: /lisa-virtualize-invoke/api/v1/vse/vsProcessor/b9e24713-eb8a-48cf-a42f-478ff61ee56b/showRequest  timeFormat: "YYYY-MM-DD HH:mm:ss z"  } | VSE |
| GET | /lisa-virtualize-invoke/api/v1/vse/vsProcessor/{processorId} | Get Virtual Service Processor Information | None | Returns above JSON or Processor Expired message in JSON | VSE |
| DELETE | /lisa-virtualize-invoke/api/v1/vse/vsProcessor/{processorId} | Remove Virtual Service Processor | None | None | VSE |
| POST | /lisa-virtualize-invoke/api/v1/vse/vsProcessor/{processorId}/showRequest | Create request object from a raw request | Raw Request bytes in request body | application/json  {  - arguments: {  cmd: "list"  }  attributes: { }  binary: false  body: null  charset: "UTF-8"  matchText: null  -  metaData: {  Accept-Language: "en-US,en;q=0.5"  HTTP-URI: /itko-examples/user-manage.jsp  Host: "[localhost:8080](http://localhost:8080/)"  Accept-Encoding: "gzip, deflate"  DNT: "1"  User-Agent: "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:31.0) Gecko/20100101 Firefox/31.0"  HTTP-Method: "GET"  HTTP-Version: "HTTP/1.1"  Accept: "text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8"  }  operation: "GET /itko-examples/user-manage.jsp"  } | VSE |
| POST | /lisa-virtualize-invoke/api/v1/vse/vsProcessor/{processorId}/findAllMatchingTransactions | Find all matching transactions for a raw request | Raw Request bytes in request body | Array of matching transaction ids. application/json  [ 5 2 ] | VSE |
| POST | /lisa-virtualize-invoke/api/v1/vse/vsProcessor/{processorId}/importRequestResponsePairs | Import request response pairs. Updates the service image with the imported transactions. Send back a model archive with the updated service image. | MultipartFile data in parameter rrfiles  The rrfiles are request/response/meta-data files that are used for importing RR pairs. | Updated Model Archive | VSE |

LikeBe the first to like this