

Introduction to XOG

Introduction

CA PPM, formerly CA Clarity PPM, is an industry-leading Project and Portfolio Management system that enables your organization to manage its services, projects, products, people, and financials. One of CA PPM's most important capability is its ability to exchange data with external applications. This exchange of data is performed through integrations and CA PPM's XML Open Gateway (XOG) and is the basic building block of all CA PPM's integrations.

With XOG, data can be exchanged with financial systems such as SAP, resource management systems such as PeopleSoft or legacy PPM applications.

In this brief whitepaper, we will explain what XOG is and describe some of the benefits and limitations of XOG. We will also illustrate how XOG works and show different ways to execute XOG. Finally, we will discuss some best practices in exchanging data with CA PPM.

What is XOG

XOG is CA PPM's Web service interface. It has been in use since 2001 when it was introduced in an earlier version of CA PPM, called Niku 6.0.

XOG uses Extensible Markup Language (XML), Simple Object Access Protocol (SOAP) and Web Services Description Language (WSDL) to exchange information with other applications. XML is the format used for the XOG requests. WSDL is used to describe the available XOG services and the format of the XOG document. SOAP is the protocol used to send the XOG requests.

Web services, XML, SOAP and WSDL are industry standards and extensively used in integrations between applications.

Benefits of XOG

The main benefit of XOG is the role it plays in how CA PPM integrates with external applications. CA PPM integrations allow companies to leverage their current application investments. Through integrations, new employees added to PeopleSoft can automatically become resources in CA PPM or new projects added to SAP can become projects in CA PPM. Even charges added to CA PPM can automatically be sent to SAP for invoicing. Using industry standard Web services, any application that can make SOAP calls is able to communicate with CA PPM in real time.

XOG also plays an important role in enhancing CA PPM to support a client's business processes. CA PPM processes use GEL to program custom scripts as part of the logic for workflows. In many cases, these GEL scripts need to create or update data within CA PPM and this is accomplished by making XOG SOAP calls.

Another benefit of CA PPM XOG is that any data imported through XOG will adhere to any business rule configured in CA PPM. If the creation of a Project in CA PPM requires the population of a custom attribute, then no project can be created through XOG without the XOG call populating that attribute.

XOG requests can be made using any programming language that supports SOAP calls. This includes Java, Perl, C#, C++, Visual Basic and GEL. Making XOG requests through SOAP calls reduces limitations to the development of your integrations and allows your company to leverage the best language for your integration.

Any item created with CA PPM Studio can be accessed through XOG. XOG can access an item's configuration information and if the item is a custom object, its data can also be exchanged using XOG.

XOG is the preferred and supported process for migrating configuration changes between CA PPM environments.

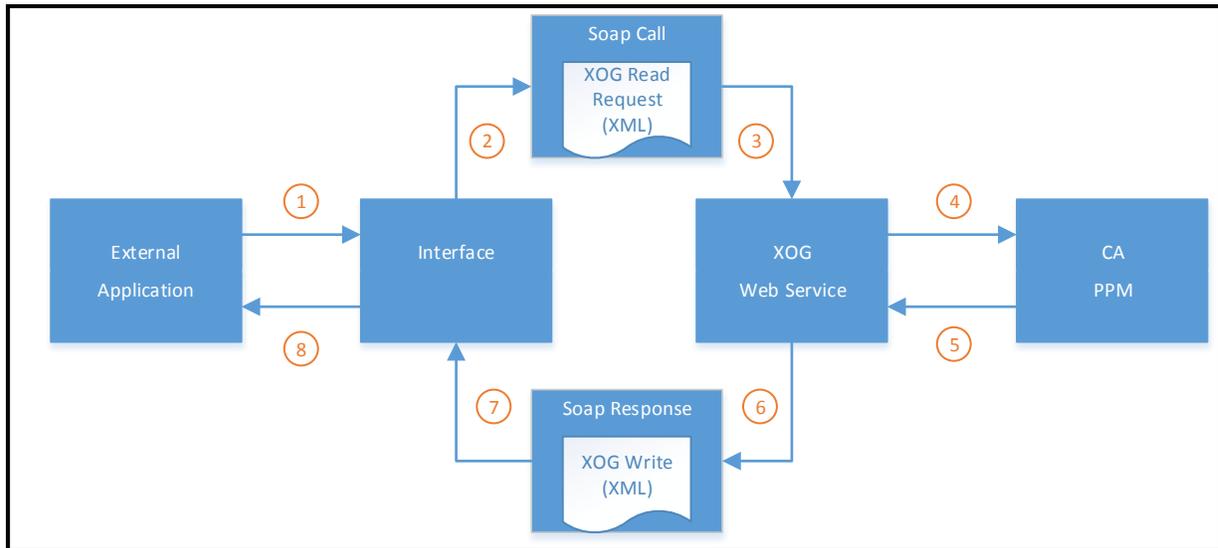
XOG is a CA supported technology. If you encounter an issue with XOG, you can open a support ticket with CA Support to determine the cause. XOG is usually upward compatible, reducing your development costs when upgrading to a newer version of CA PPM.

XOG is secure and rights-enabled. A user must be properly authenticated and have the appropriate rights to the data in order to access or modify the data through XOG Web services.

Clients with On Demand implementations of CA PPM are only granted read-only access to their CA PPM database. XOG Web services is the only permissible way to create or edit data within a CA PPM On Demand implementation.

How XOG Works

As a Web Service, XOG requests are invoked using SOAP calls. XOG requests are XML messages that comply with the SOAP protocol. XOG requests may be invoked from any application that supports SOAP calls.



Request Data from CA PPM

- Step 1 - The external application requests data from the interface
- Step 2 - The interface generates a XOG Read Request and makes a SOAP call
- Step 3 - The SOAP call is sent to the XOG Web service
- Step 4 - The XOG Web service validates the requests and requests the data from CA PPM
- Step 5 - CA PPM returns the requested data to the XOG Web service
- Step 6 - The XOG Web services formats the data from CA PPM into a XOG Write requests and generates a SOAP response
- Step 7 - The SOAP response is sent back to the interface
- Step 8 - The interface processes the data from the SOAP response and returns it to the external application

The illustration above describes the steps a simple interface might take when requesting data from CA PPM.

Limitations of XOG

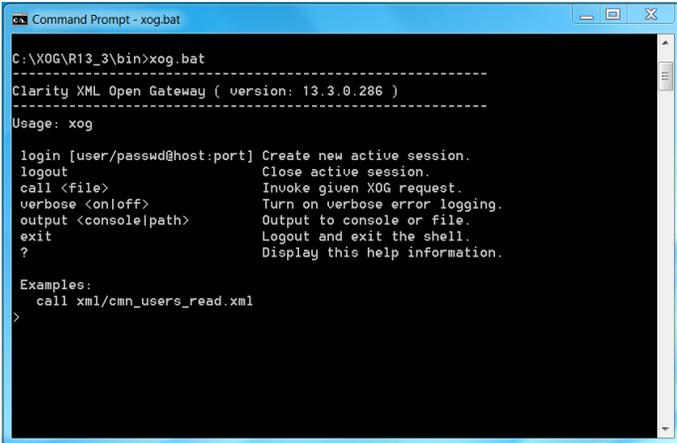
With few exceptions, XOG requests can only create or edit data in CA PPM; XOG typically will not allow a user to delete data. The exceptions include the XOG write requests for OBS Associations, Skill Associations, Group Assignments, Global Rights, Instance Rights, OBS Rights and Rate Matrices. In these cases, you can replace a set of existing records with a new set (for example, if resource John Doe has rights A, B and C, you can send a XOG write request with only rights A and C and effectively remove right B from the resource).

There is a limit to the granularity of a XOG export. As an example, to XOG out a task, you need to XOG out the project the task belongs to. In the same note, if you want to XOG in a task, it must be part of a project XOG write request.

Performance of XOG requests is exponentially affected by the number of records being imported or exported. Large XOG requests can take hours to process, and very large requests can time-out the XOG session. Starting with version 14, XOG will paginate the number of records processed, reducing the possibility of timing out, but this requires special logic in the integration to handle the pagination.

Executing XOG Requests

CA PPM provides a XOG client that can be executed in a Unix or Windows environment. The XOG client can be run in “shell” mode or as a script (batch file) command. The shell mode is useful when testing your XOG requests. The script mode is useful when performing multiple XOG requests.



```
Command Prompt - xog.bat
C:\XOG\R13_3\bin>xog.bat
-----
Clarity XML Open Gateway ( version: 13.3.0.286 )
-----
Usage: xog

login [user/passwd@host:port] Create new active session.
logout Close active session.
call <file> Invoke given XOG request.
verbose <on|off> Turn on verbose error logging.
output <console|path> Output to console or file.
exit Logout and exit the shell.
? Display this help information.

Examples:
  call xml/cm_users_read.xml
>
```

XOG Client in “shell” mode

XOG requests can also be executed through SOAP calls. SOAP calls can be executed through Java, .Net, C++, perl, and other programming languages. GEL is also able to make SOAP calls.

Integration Considerations

As mentioned earlier, any language that can make and receive SOAP calls can be used to develop integrations with CA PPM. This frees the client to use the language most suitable to their development.

There are many factors that need to be considered when developing an interface with CA PPM, such as,

- “What data needs to be exchanged?”
- “How does the data map to CA PPM?”
- “How often does the dataset change?”
- “Should it be scheduled or real-time?”
- “How should the interface kick off?”
- “Is CA PPM On Premise or On Demand?”
- “Do the results need to be audited?”

- “What type of notifications are required?”
- “Who should be notified of success or failure?”
- “Who should have access to the interface?”
- “Are there any limitations that need to be considered” or
- “Who will be maintaining the code?”

Addressing these questions during requirements gathering for the interface will give the developer better insight into the client’s needs.

Another important consideration when developing an interface is whether the data needs to be processed as a whole, or can some records be processed when others fail. Developing an interface that necessitates all-or-nothing processing requires the interface to pre-process all the data before attempting to exchange the data with CA PPM. This adds complexity and processing time to the interface. This condition usually only involves scheduled interfaces, as real-time interfaces typically deal with individual records.

Inevitably when developing an interface with CA PPM, there will be data from the external application that is currently not being represented in CA PPM. How this data is configured in CA PPM can have an effect on how well the interface can handle the data exchange. For example, dynamic lookups must have NSQL queries that follow specific guidelines to ensure data exchange quality. This is described in the Best Practices section.

When using a data file to interface with CA PPM, the standard practice is to use a delimited ASCII file. Consider using a character other than “,” (comma) as the delimiter as it is usually found in the data of attributes. Try using a less common character as the delimiter.

Find a good balance between performance and maintainability. A fast integration that cannot be easily maintained can end up being costly when changes are needed.

XOG Best Practices

When working with XOG, it is important to keep your requests as small as possible, i.e. do not request or send too much data at once. Many smaller requests will usually execute faster than one large request.

XOG in only what is necessary.

Whenever you XOG data out, review it for the inclusion of XOG dependencies. This is important when you XOG out configurations from Development, as unintended changes could be included in the XOG file. For example, when executing a XOG read request for a Custom Object configuration, if the object is partitioned, then the partition model will be included in the XOG results. If you are not intending to make changes to the partition model of the destination environment, then make sure you remove the partition model from the XOG file before executing the XOG file in the destination environment.

XOG out only what you need. Many XOG read requests have filters that can be used to limit what information is returned. This reduces the complexity of the resulting XOG file and improves performance.

When defining dynamic lookups in CA PPM, define a “unique_name” value in the NSQL. Including this value in the Lookup NSQL tells the XOG Web service to use that value when exchanging data with CA PPM. This will prevent XOG from using an internal key value in the XOG data exchange.

Make sure to specify an ID when defining project tasks. This provides a cleaner project XOG, especially when dealing with assignments.

Summary

The addition of XOG to CA PPM has been a benefit to clients. As CA PPM’s data exchange mechanism, it has allowed clients to extend their software investments, use best-of-class applications for different activities, create a gradual transition into CA PPM or migrate legacy data into CA PPM. Using XOG with GEL in CA PPM processes has allowed clients to enhance their CA PPM implementation to better support their business model.

The foundation of any integration with CA PPM is XOG, and understanding its strengths and weaknesses will help you design and develop an effective integration. In general, keep your XOG requests small and simple, make sure your dynamic lookups are XOG-friendly, test you XOG using the XOG client, make sure to enter task IDs and only XOG in what you intend to add or change.

Following these basic guidelines to develop your interfaces will enhance your investment in CA PPM.

WinMill Software – We Make Things Happen!

WinMill Software is a CA Global Channel Partner and Preferred Services Partner, and the leading provider of CA PPM consulting services in North America. WinMill has been a CA Alliance Partner since CA acquired Niku in 2005, and prior to that was a Niku Alliance Partner. All WinMill CA PPM consultants are CA certified. We provide expert technical and functional CA PPM consulting services, as well as CA PPM project management. Contact us at 888-711-MILL (6455) or inquiry@winmill.com.

CONNECT WITH US

