

Custom Process Model (CPM) Best Practices

Overview of CPM best practices, including concepts around the appropriate implementation of API suppression, exception/error handling and logging

Chris Hunter, Bug Diagnostic Engineering - Customization Product, Global Product Support, OSvC

Helpful topics to understand

API Suppression

- API suppression in code
- API suppression configurations in Process Designer
- How are rules, campaigns and data imports involved?

Exception/error handling

- Appropriate Exception types to use
- What are the different considerations when for sync vs. async CPMs

Logging and Mail API notification

- When is it appropriate to use CPM logging?
- When is it appropriate to use Mail API notification?
or both?

The successful use of these tools will result in fewer production problems and maintainable customizations over time.

API suppression

In code

- Is API suppression indications within API save statement enough?
 - No, be sure to use the 'Can Suppress' check box configurations made in the Process Designer also
- API saves can suppress business rules, CPMs, or both

In Process Designer

- When do I need to ensure the 'Can Suppress' check box is checked within the Process Designer?
 - If you have CPMs mapped to be triggered for objects mapped in the Process Designer, and intend to have API saves in code with suppression of CPMs
 - Be sure changes are deployed

Rules, campaigns and data imports

- Business rules can be used to trigger CPM customizations
- Campaigns can be used to trigger CPM customizations
- Data imports from the Data Import Wizard can be used to trigger CPM customizations

Exception/error handling

Appropriate exception types to use

- `\Exception`
 - Use backslash for escaping the CPM namespace and the standard `Exception` class will catch all types of exceptions
- CPHP exceptions are available also (`ConnectAPIError`, `ConnectAPIErrorFatal`)

Considerations for synchronous CPMs

- Unhandled exceptions in synchronous CPMs will cause the server process (depending on source such as console/public API/CPM) to halt
- Handling exceptions within CPM code will allow processing to continue, but care should be made that maintainers of the site will be aware the error has occurred

Considerations for asynchronous CPMs

- Asynchronous CPMs are run using a site utility that processes a queue
- Asynchronous CPMs queued will be re-tried when an uncaught exception is encountered
 - It is recommended to catch exceptions, and then manually throw an exception in code if you want the CPM to be re-tried

Logging and Mail API notification

- The CPHP Mail API is intended to be used to notify the maintainers of site customizations when problems arise
- CPM logging should be implemented so that the cause of problems can be determined retroactively (vs. trying to find a way to reproduce the problem, and then debugging once that is found)

Other best practices and gotchas to be aware of

- When importing classes with CPMs always use a backslash to escape the CPM namespace
- When buttons are clicked in the Process Designer, be sure to wait for the operation to complete (notice “Sending” indication in bottom-right corner of the console)
- Use prescribed steps when deleting CPMs

- The Connect for PHP API is already initialized and authenticated behind the scenes in a CPM customization
- Always use a timeout with PHP curl in asynchronous CPMs (value of 5 seconds is recommended)
- Also when using PHP curl, check that it is not already loaded before running the `load_curl` function

- Custom Processes must have matching version information between the annotation and included Connect library
- CPMs should be developed and tested on a test site, or clone of a production site, before implementing on a production site
- It is always best to use the latest CPHP framework version

Where can I find further information on these topics?

CPM/Process Designer Best Practices and Guidelines

https://cx.rightnow.com/app/answers/detail/a_id/8392

Connect for PHP Best Practices and Gotchas

https://cx.rightnow.com/app/answers/detail/a_id/9640

Adding custom logging to your CPHP customizations

https://cx.rightnow.com/app/answers/detail/a_id/9897

Enabling API suppression in Connect for PHP Customizations

https://cx.rightnow.com/app/answers/detail/a_id/7890

When does a Custom Process (CPM) enter an error state?

https://cx.rightnow.com/app/answers/detail/a_id/1002

Mail API per-process limit of 200 emails

https://cx.rightnow.com/app/answers/detail/a_id/10102

Custom Process Model (CPM) Defect Analysis

https://cx.rightnow.com/app/answers/detail/a_id/9750

The Process Designer / Custom Process Models (CPM) functionality

https://cx.rightnow.com/app/answers/detail/a_id/6246

Anatomy of a CPM (Custom Process)

https://cx.rightnow.com/app/answers/detail/a_id/10086

Asynchronous CPM "processes" queued are removed automatically after five unsuccessful tries

https://cx.rightnow.com/app/answers/detail/a_id/6607

Continued:

Asynchronous CPM customizations triggered on one interface can run on any enabled interface

https://cx.rightnow.com/app/answers/detail/a_id/10377

Two second guideline for synchronous CPMs

https://cx.rightnow.com/app/answers/detail/a_id/9453

Custom Processes must have matching version information between the annotation and included Connect library

https://cx.rightnow.com/app/answers/detail/a_id/8655

Use of global variables in Custom Process object event handler scripts

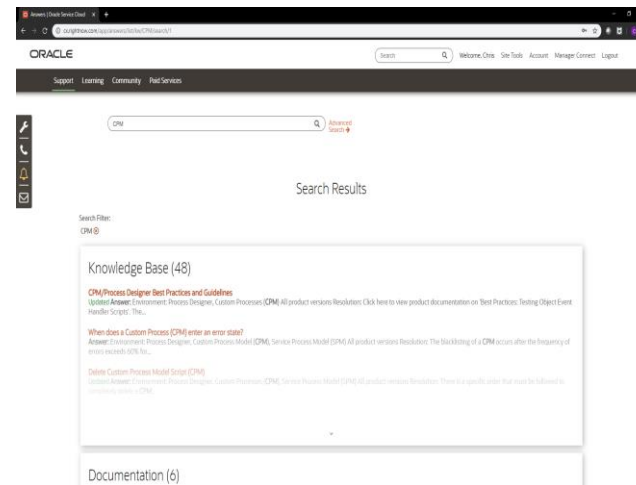
https://cx.rightnow.com/app/answers/detail/a_id/6701

There are two types of execution for Object Event Handlers

https://cx.rightnow.com/app/answers/detail/a_id/6604

Where is the best place to find this type of information on my own?

- Support site knowledge base
- Product feature documentation
- Service Cloud Community
 - all of this information is available if you login and search at cx.rightnow.com



Questions?