



Web ADI vs. More4Apps Wizards

More4Apps sometimes gets asked by customers and prospects for a comparison between Oracle Web ADI and More4Apps products. This document has been put together to clarify the core differences between the functionality of these two desktop integration solutions.

Seeded Web ADI Tools and Custom Web ADI Integrators

For some areas within Oracle Applications there are a few seeded Web ADI Integrators available. Examples include: GLDI and the Salary Proposal Integrator. These are pre-configured and generally do a reasonable job of integrating between Excel and Oracle. Usually More4Apps do not develop similar tools to existing Web ADI Integrators (except in the circumstance where it has been driven by customer-demand).

There are plenty of situations where no seeded Web ADI Integrator is available and in those circumstances you may look to create your own custom Web ADI Integrator. Using Oracle functionality a custom Web ADI Integrator can be set up to call one procedure within one package (generally an Oracle API). This can then be used to load transactions between Excel and Oracle.

For the purpose of this document we are comparing creating your own Custom Web ADI Integrator vs. implementing a More4Apps Wizard.

Custom Web ADI Integrator(s) vs. More4Apps Wizard(s)

Subject / Function	Web ADI Integrators	More4Apps Wizards
Implementation, Configuration and On-going Support	<p>A large number of set up steps. Heavy reliance on functional expertise and technical knowledge.</p> <p>E.g. the knowledge and ability to complete all the functional set up steps; understanding of the API tables, packages and procedures available; the ability to create lists of values and defaults.</p>	<p>Wizards are pre-configured.</p> <p>A DBA will need to install the two More4Apps packages into the database. A functional person will need to undertake some testing and possibly modify the worksheet layout to suit the business purpose.</p>
Values Displayed When populating the worksheet with bulk information it is more practical for a user to be working with values rather than database ID's.	<p>Sometimes</p> <p>Web ADI will display the fields as they are listed in the API package. Which means you may be required to provide ID's rather than the user-friendly values.</p> <p>There is a technical work-around, you can have a developer write a custom "wrapper package" that translates the values into ID's and then calls the standard API package. In this instance the Web ADI Integrator would be set up to call the custom package.</p> <p>This work-around increases the reliance on technical expertise.</p>	<p>Yes</p> <p>Displays values to the user and if necessary will translate values into ID's prior to calling the API.</p>

Master-detail Type Relationships Supported	<p>No</p> <p>You can only call one API procedure within one API package per integrator. So for example, if you need to load Employees with Addresses, Assignments, and Salaries, you would need to set up multiple integrators and use multiple spreadsheets.</p>	<p>Yes</p> <p>More4Apps Wizards have a spreadsheet layout that can present master-detail type relationships.</p> <p>Multiple API's can be called within one wizard.</p>
Attachments	<p>No</p> <p>You cannot load attachments while loading the related transaction using a Web ADI Integrator.</p>	<p>Yes</p> <p>Many More4Apps Wizards have been enhanced to include Attachments functionality (as customers have requested the need for it). E.g. AP Invoice Wizard, P.O. Wizard, Requisition Wizard.</p> <p>More4Apps continues to add Attachments functionality into Wizards where customers have a need for it.</p>

More4Apps Wizards

Some key points...

- More4Apps Wizards are pre-configured and require minimal functional and technical expertise to get running on your instance. A DBA would be required to run two scripts to install packages into your database. A functional person will need to do some testing and decide on a worksheet layout suitable to your business purpose(s).
- More4Apps Wizards can handle master-detail relationships within one spreadsheet. They can call multiple API's, meaning one wizard can be used to load parent and related child records.
- More4Apps Wizards expect the user to enter "Values" not the database ID's. Meaning that the user can understand the information they are populating in the worksheet.
- More4Apps Wizards provide forms that are configured automatically to the instance they are connect to. No additional set up is required; the lists of values, flexfields etc contain the values the user will be expecting.
- The majority of the More4Apps products provide create, download, validate and update functionality within the one Wizard.

Seeded Web ADI Integrators

Some things to look out for when considering seeded Web ADI integrators...

- Are list of values provided? Are the users expected to know and enter database ID's? If yes, this can make data-entry difficult.
- Are relevant descriptive flexfields available within the worksheet?
- The 'Create' and 'Update' functions are likely to be separate; generally a Web ADI integrator will only do one or the other. If you need to do updates, do you have the ability to download/extract information and make changes to it?
- Web ADI generally does not handle master-detail relationships, so you are likely to be able to create records for one entity and then have to use additional integrators to populate child/related records. Consider if this is practical and how it will impact on productivity.

As per usual... be sure to test various scenarios thoroughly and ensure that the product is suitable for the tasks you will be performing.

Creating your own Web ADI Integrators

There are many situations where there are no seeded Web ADI Integrators available. Here are some things to look out for when creating Custom Web ADI Integrators (using standard Oracle functionality)

- Web ADI integrators take a significant amount of set up and testing, you are likely to need a reasonable amount of functional expertise and at least some technical knowledge (e.g. the knowledge and ability to complete the functional set up; some understanding of the API packages and procedures available; the ability to create lists of values and defaults).
- Web ADI integrators can only be created for one API procedure at a time. For example you could create an integrator for an employee record calling 'HR_EMPLOYEE_API.create_employee', but you would need to set up another integrator to do the update.
- Web ADI integrators will not handle parent-child relationships within one integrator. In the example above you would need additional integrators and spreadsheets to create an employee's address, another for a contact, another for a phone number, another for an assignment and so on. So how many are you going to need to set up, use and support?
- Web ADI integrators will display the parameters within the procedure being called (as fields within the worksheet). An API will often require many ID's to be passed into it. In practical terms a user will want to be entering the values rather than the database ID's. If you want to convert user values in the spreadsheet into ID's for the API, then you will need a developer to write a wrapper package that does this conversion and then calls the API procedure itself. Significant development, technical and functional expertise is likely to be required to implement this.