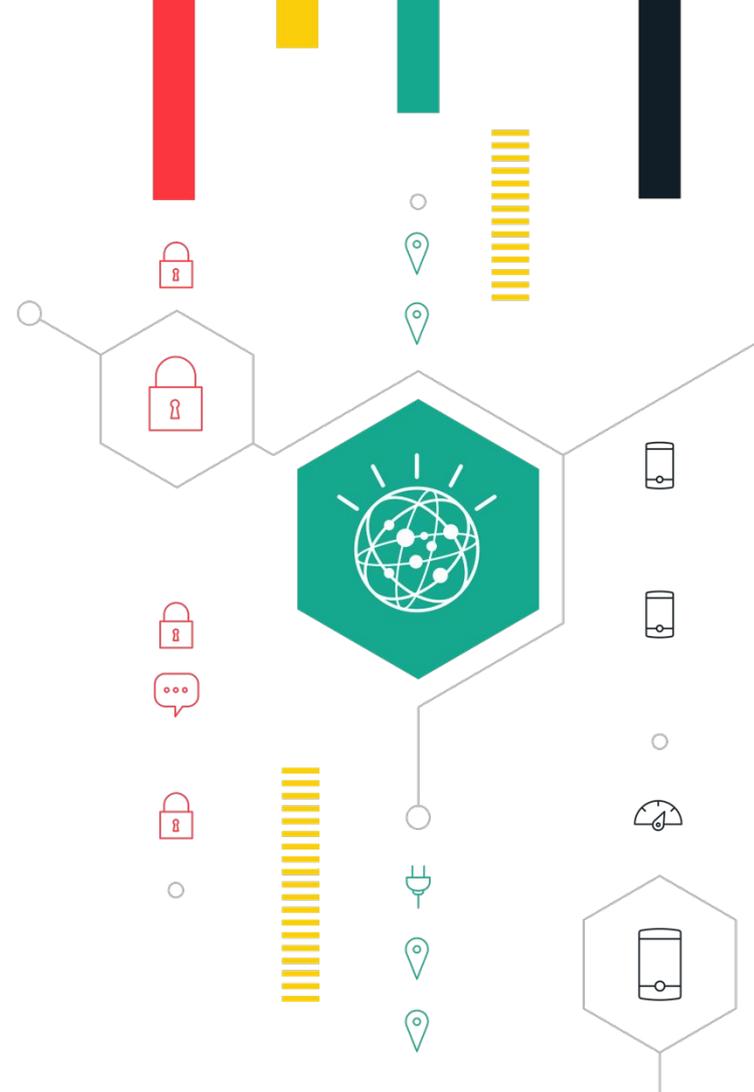


IBM **Watson IoT**

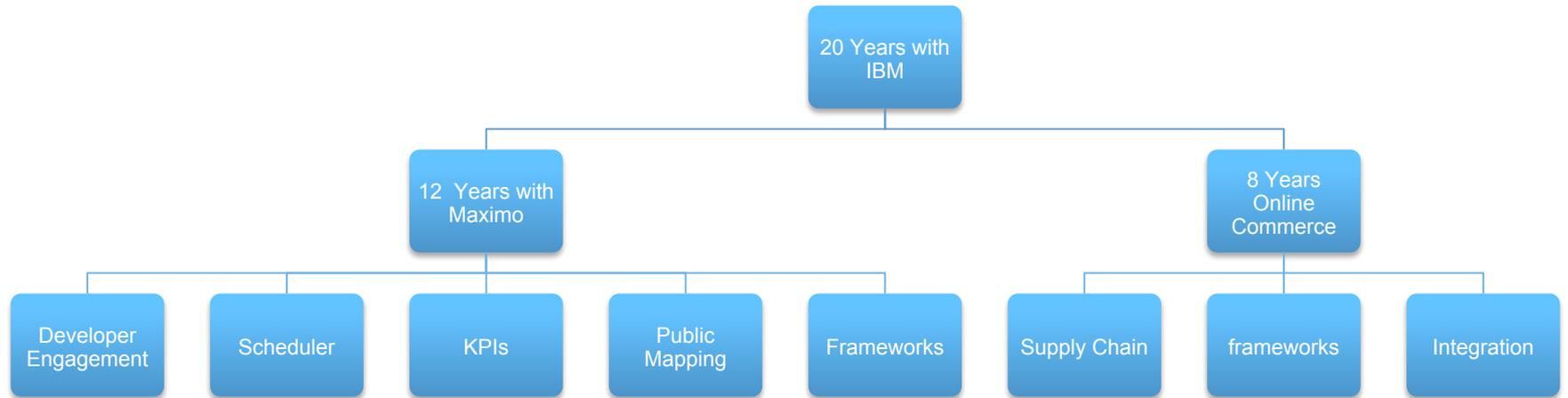
IBM Maximo Customization Scripting and Formulas

Oct 2018

Sean Stuckless
sls@ca.ibm.com



Who is this guy?



DISCLAIMER

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

© Copyright IBM Corporation 2018. All rights reserved.

Scripting and Formulas

Maximo thrives on customization and configuration

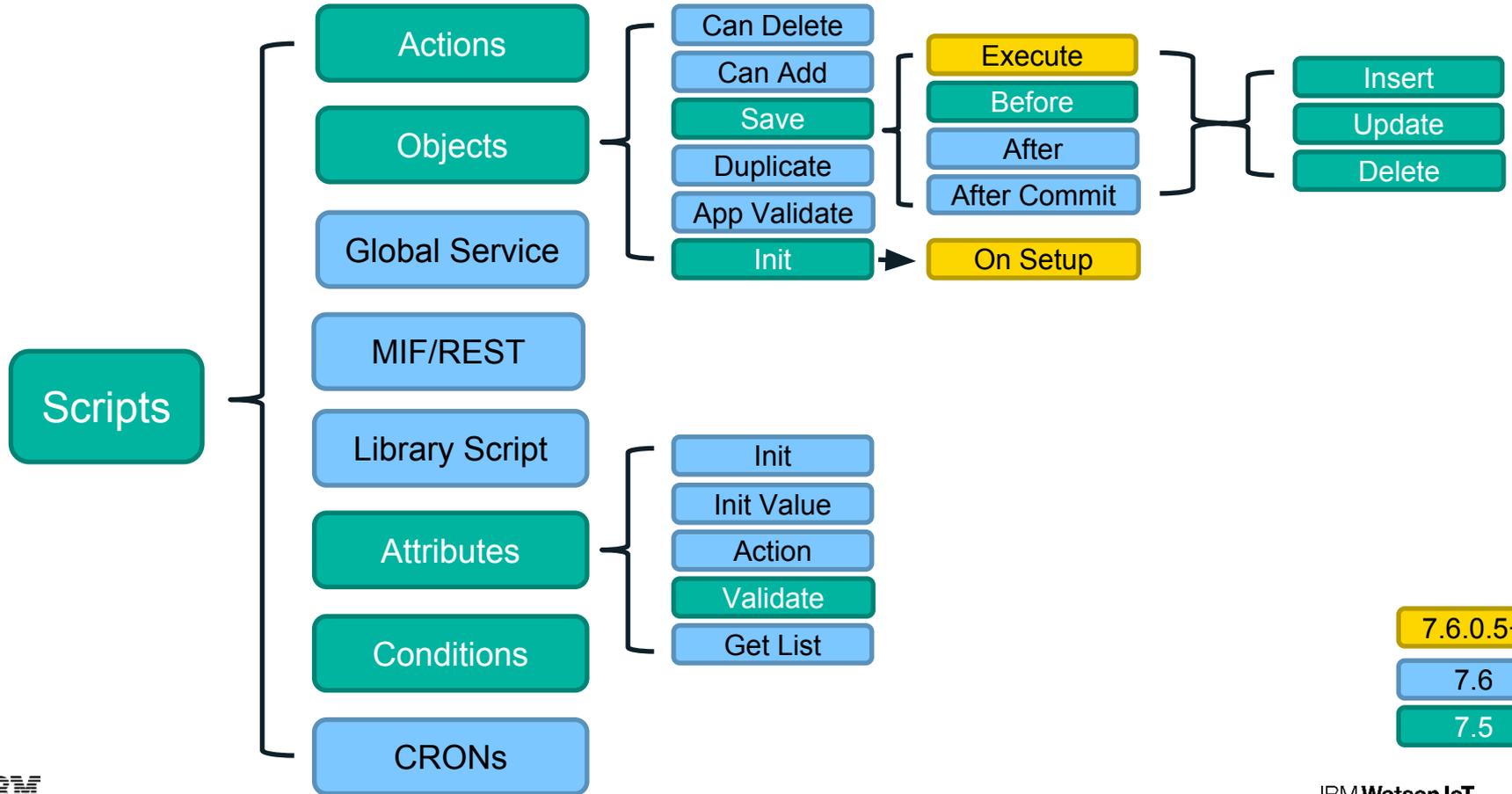
Fast and dynamic(if used properly)

Lots of integration points

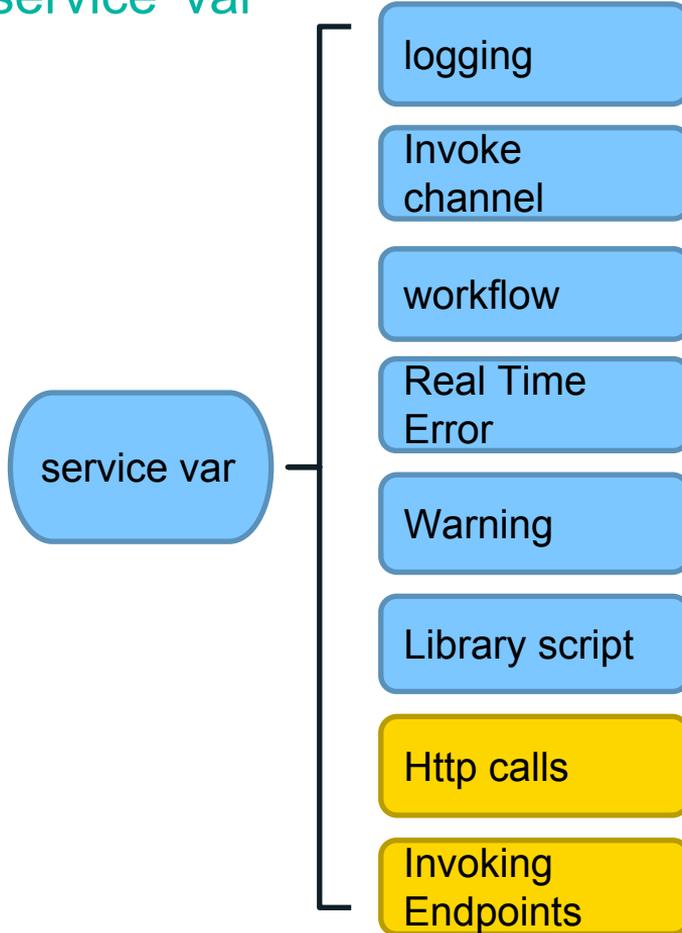
Calculated fields using Formulas is natural

Workflow is OK, but, bulky for extending business logic

Scripting and Formulas



Global 'service' var



7.6.0.5+

7.6

7.5

Service Examples

```
service.error(grp, key)
```

```
service.error(grp, key, params)
```

```
service.setWarning("po", "nolines", None)
```

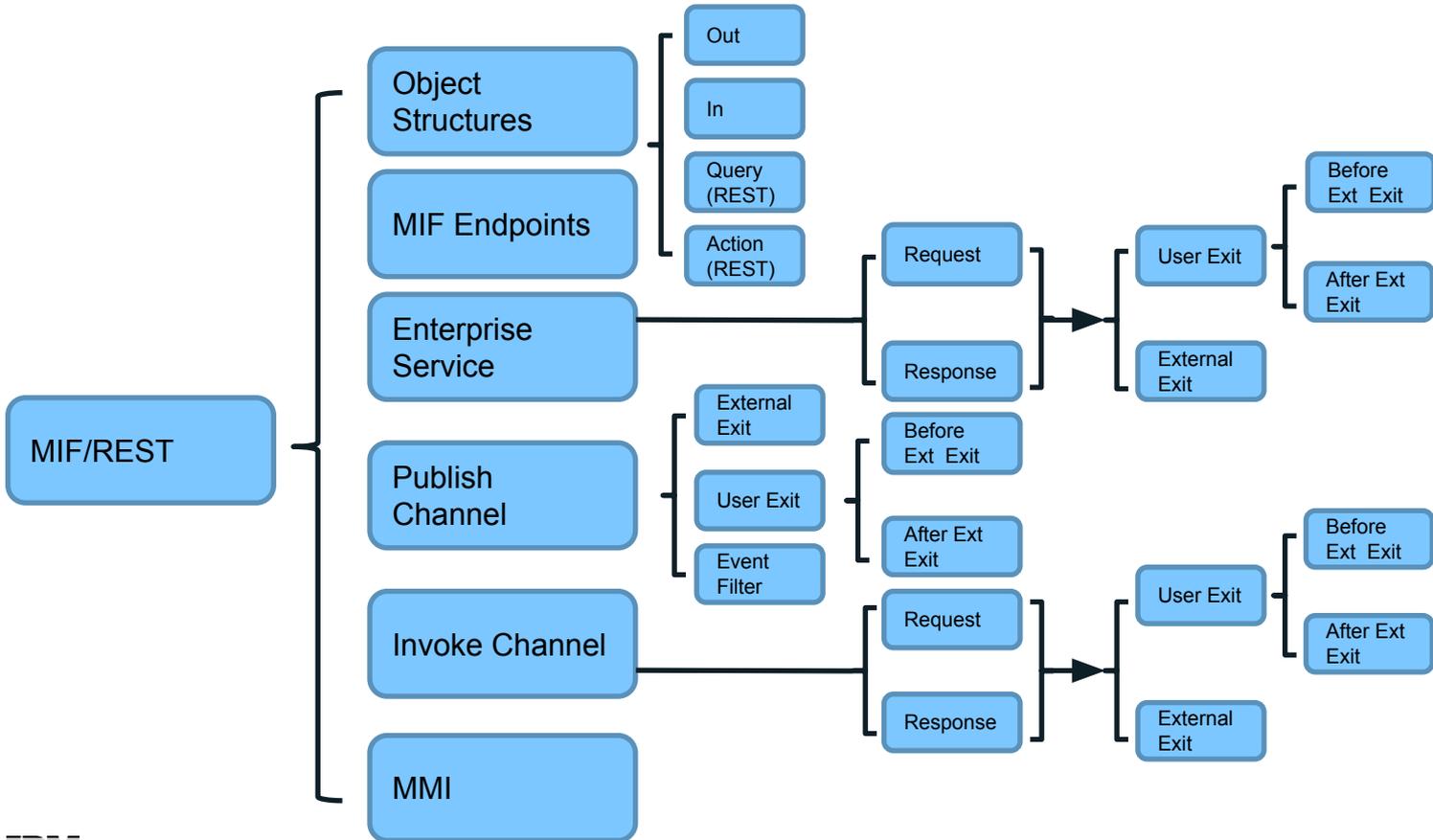
```
service.yncerror("asset", "assetpr", params)
```

```
service.logError("an error happened")
```

```
service.invokeChannel(channelname)
```

* service in `ctx` in MIF OS Scripts

MIF/REST Scripting Points - Exits

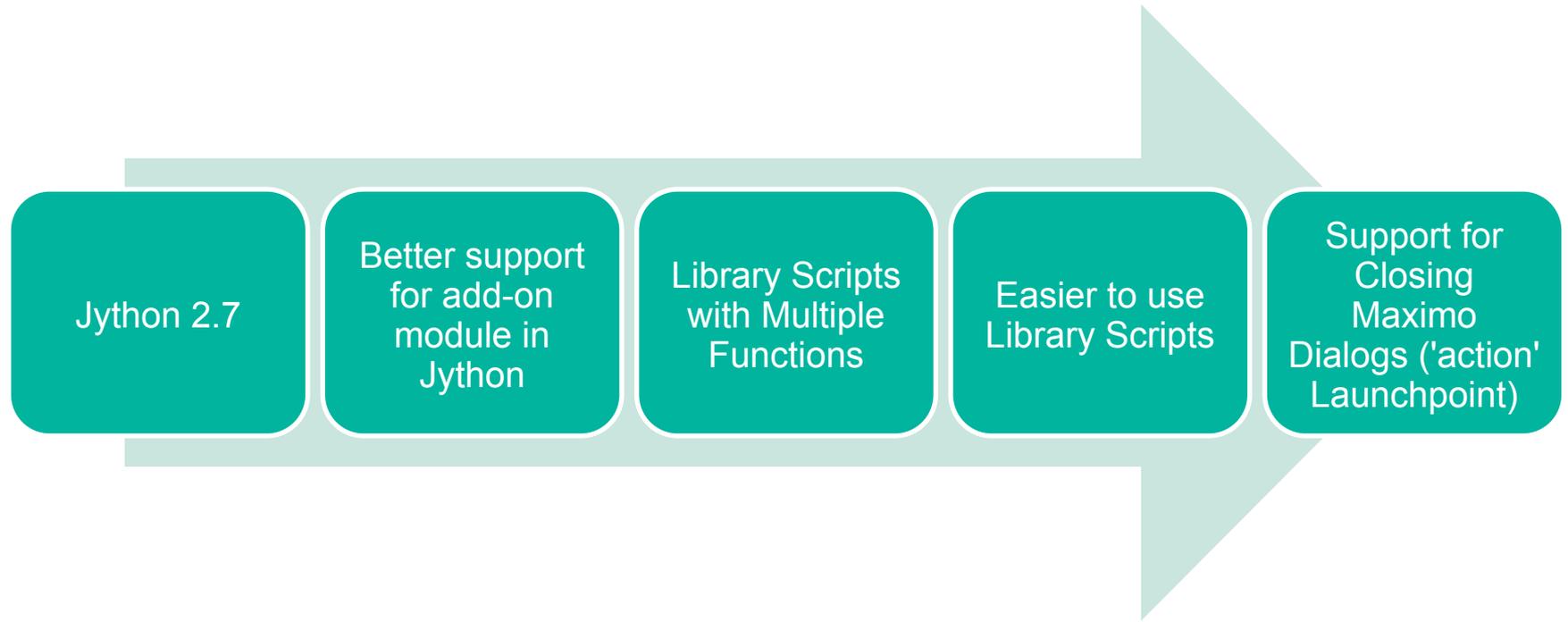


7.6.0.5+

7.6

7.5

New in 7.6.1



Reusing Scripts – Library Scripts

- ❖ Current way to write Library Scripts.

Script 1 – CALC_MULT

$r = a * b$

Script 2 – CALC_ADD

$r = a + b$

Reusing Scripts – Library Scripts

- ❖ Current way to consume such a Library Script.

```
from java.util import HashMap
map = HashMap()
map.put("a", 2)
map.put("b", 3)
service.invokeScript("CALC_MULT", map)
res = map.get("r")
```

Reusing Scripts – Library Scripts

- ❖ What you can do now (make sure that you set the script to **Allow Invoking Script Functions**).

Script - CALC

```
def mult(a,b):  
    return a*b
```

```
def add(a,b):  
    return a+b
```

Reusing Scripts – Library Scripts

- ❖ The script to consume it.

```
res = service.invokeScript("CALC", "mult", [2, 3])
```

Scripting Performance and Common Problems

- Not closing cursors
- Full access to Java and Maximo
- Too many relationships
- Too much database access
- Calling expensive methods (Change Status)
- Incorrect Lifecycle Event
- Calling "Save"
- Complex conditional Scripts

Dev Center + Future

- One Stop - Samples / References
 - <https://github.com/ibm-maximo-dev/maximo-scripting>
- Auto Script Editor
 - <https://tinyurl.com/maximo-autoscript-editor>
 - Eventually will rolled into Maximo
- Auto Script Revisioning (Possible enhancement)

Formulas

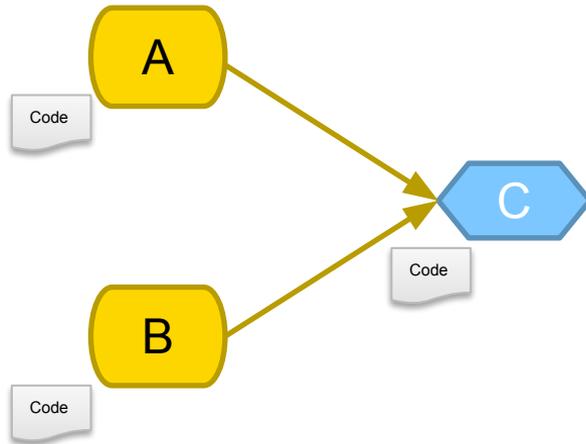
Formulas

- Calculated Fields - Mathematical expressions
- Excel Like (Built-in functions)
- Access Related Fields
- Aggregation
- Applied to Objects and Attributes
- Extensible Function Framework in Java or Automation Scripting
- Easy Access from DB Configure Application
- Persistent or Non Persistent
- Immediate or Asynchronous

7.6.0.6+

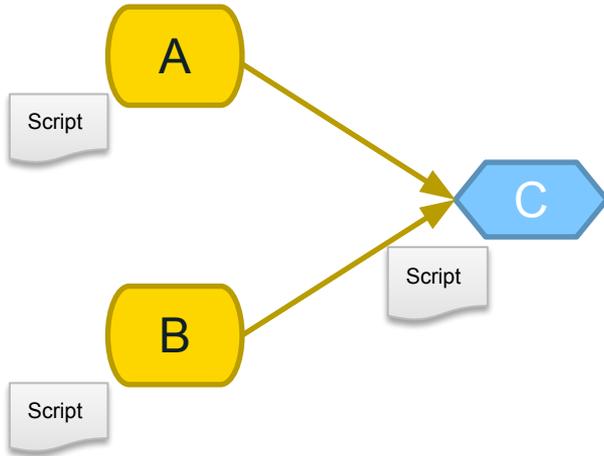
Formulas vs Scripting vs Java

$C = f(A,B)$ - Java



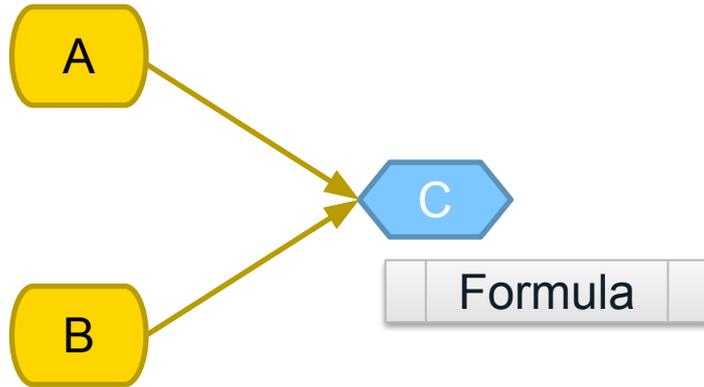
- Code Attached to Each Field
- Different Lifecycle for each Field
- Rebuild Maximo Ear

$C = f(A,B)$ - Scripting



- Script Attached to Each Field
- Different Lifecycle for each Field
- Configure Inputs/Outputs

$C = f(A,B)$ - Formula



- Single Function
- Configured in Database Application
- C auto-magically updated when A or B Change

Formulas in Maximo

- Approx 30+ Functions as of 7.6.1
- min, max, round, pct, if, date, abs, pow, sqrt, ...
- Including aggregation functions, avg, sum, ...
- Can Reference
 - Mbo attributes
 - Maxvars
 - System Properties
 - Conditions
 - Other Formulas

Example: Using Conditional Logic

```
IF (pressuremeter$lastreading > 60, 1,  
    IF (pressuremeter$lastreading > 20  
        && pressuremeter$lastreading < 59, 2, 3) )
```

- Nested Logic
- Relationships using the \$ operator
- Implicit type conversion from ALN to double, boolean, date

Formulas - Prefixes

Prefix	Description	Example
prop	System Property	<code>prop\$propname * 1.5</code>
var	MAXVAR	<code>IF(var\$maxvarname, 1, 2)</code>
cond	Maximo Condition	<code>field * IF(cond\$conditionnum, 0.25, 0.75)</code>
isnull	In Null Test	<code>IF(isnull\$installdate, 0, 1)</code>
prev	Previous Value	<code>IF(priority > prev\$priority, 1, 0)</code>
modified	Is Modified Test	<code>IF(modified\$priority, 1, 0)</code>

Formulas – Aggregation and Collections

Prefix	Description	Example
SUMF	Sum	<code>SUMF("childtask", "estdur", "actfinish", DURATION(0,0,7,0,0,0))</code>
COUNTF	Count	<code>COUNTF("openwo", "statusdate", DURATION(0,0,10,0,0,0))</code>
oldest	Oldest by Date	<code>oldest\$openwo\$wopriority\$statusdate</code>
latest	Newest by Date	<code>latest\$openwo\$wopriority\$statusdate</code>
count	Row count	<code>count\$childtask</code>

➤ AVGF, MINF, MAXF also available

When to use Formulas vs Scripting

- Formula
 - Mathematical Calculation
- Scripting
 - Validation
 - Complex Updates (if then else)
 - Business Logic

Formula Performance and Common Problems

- Use aggregation sparingly
- Too many relationships
- Badly written user defined Functions
- Complex Conditional Formulas

THANK YOU