



# Monthly Publishing System (MPS) Developer Workshop

25 August, 2015

# High-level Overview

- Introduction
- Preliminary discussion session
- Overview of the MPS
  - Lunch
- Presentation and Workshop
- Wrap-up discussion session
  - Finish ('Break' for those that want to continue)
- Informal Workshop

# Preliminary Discussion Session

- Results of the Developer Survey

- Design Objectives



# Monthly Publishing System (MPS) Developer Workshop

25 August, 2015

# Workshop Agenda

- Deliverables
- Quality
  - Validation (testing)
  - Verification (counting)
- Tracking Changes
- Options for formatting
- Options for querying

# Workshop Format

- **Keep it interactive!**
  - The learning should go both ways.
  - Some questions may need to be deferred till the informal session.
- The objective is to be ‘hands-on’ but the format may require some flexibility.

# MPS Overview

- What does it deliver?
  - Reports
  - Publications
- How does it deliver?
  - MPS API
    - Summary of Changes
    - SQLite
    - JSON



# What does the MPS deliver?

- Publications
  - Summary of Changes
  - Hardcopy schedules (PDF)
  - Legislative Instruments
  - Website / offline website / mobile website
  - SQLite / SQL script
  - Text files
  - API

# Key Challenges

- **Grouping**
  - Hardcopy vs legislative instruments vs ...?
- **Sorting**
  - The primary specification is the 30 year old hardcopy.
- **Ordering**
  - Varies – a recent example is notes to restrictions.
- **Identifying**
  - The lack of permanent IDs is not optimal.

*The fundamental issue is coping with the complexity of the data in the context of each deliverable!*

## Other challenges

- The schema is big and there are a lot of places for data to hide.
- No one asks what will work best for developers before they amend the legislation.
- The first day of the month is non-negotiable.
- IE 6...

# The solution

- Both ends are volatile.
- Neither end is under our control.
- The business logic is expensive and fragile therefore duplication must be avoided.
- Stability is needed somewhere, so the only option for the MPS is to create an API in between the source data and the deliverables.

- **Start with the PBS XML and extensive validation.**
  - Well-formedness
  - Schema
  - Schematron
  - Internal format validation
- **Validation isn't enough.**
  - No errors doesn't guarantee no problems.
  - Validation errors aren't always easy to interpret.

- Count everything (between months and each version)
  - Generic Drug
  - PBS Item
  - PBS Product
  - Schedule
- Reports:  
[https://dev.pbs.gov.au/docs/reference/pbsxml\\_reports.html](https://dev.pbs.gov.au/docs/reference/pbsxml_reports.html)

# MPS Statistics Report

Effective Date: 2015-09-01

file: pbs-mps-2015-09-01-v1.xml

Total summary

Generic Drugs	Items	Brands
964	3853	6761

Drug type summary

Drug Types	Drugs	Items	Brands
CA	41	83	95
CT	11	36	60
DB	28	34	57
GE	697	2492	5034
GH	1	91	30
HB	73	235	247
HS	73	235	247
IF	12	24	24
IN	44	69	181
IP	44	69	181

Effective Date: 2015-09-01

file: pbs-mps-2015-09-01-v2.xml

Total summary

Generic Drugs	Items	Brands
963	3844	6750

Drug type summary

Drug Types	Drugs	Items	Brands
CA	41	82	94
CT	11	36	60
DB	28	34	57
GE	697	2490	5030
GH	1	91	30
HB	72	232	244
HS	72	232	244
IF	12	24	24
IN	44	69	181
IP	44	69	181

# MPS Summary of Changes

```
<changes xmlns="http://api.pbs.gov.au/0.3" effective-date="2015-08-01">
  <additions>
    <addition type="item">
      <item code="10375C"/>
      <item code="10376D"/>
      <item code="10377E"/>
      <item code="10378F"/>
      <item code="10380H"/>
      <item code="10381J"/>
      <item code="10382K"/>
      <item code="10383L"/>
      <item code="10384M"/>
      <item code="10385N"/>
    </addition>
  </additions>
</changes>
```



# MPS Summary of Changes

```
</item code="10420K" />
<item code="10422M" />
<item code="10423N" />
</addition>
<addition type="brand">
  <brand code="363651000144100" item-code="10375C" />
  <brand code="346921000144101" item-code="10376D" />
  <brand code="346911000144107" item-code="10377E" />
  <brand code="358791000144102" item-code="10378F" />
  <brand code="363991000144103" item-code="10380H" />
  <brand code="13355011000036109" item-code="10381J" />
  <brand code="926215011000036104" item-code="10381J" />
  <brand code="316441000144106" item-code="10382K" />
  <brand code="13355011000036109" item-code="10383L" />
  <brand code="926215011000036104" item-code="10383L" />
  <brand code="337361000144107" item-code="10384M" />

```

# MPS Summary of Changes

```
</deletions>
]
]
<deletions>
  <deletion type="item">
    <item code="10269L"/>
    <item code="10270M"/>
    <item code="10296X"/>
    <item code="10324J"/>
    <item code="10346M"/>
    <item code="10362J"/>
    <item code="2873F"/>
    <item code="2987F"/>
    <item code="5621W"/>
    <item code="5622X"/>
    <item code="5623Y"/>
    <item code="9157Y"/>
    <item code="9158B"/>
    <item code="9159C"/>
    <item code="9625N"/>
    <item code="9626P"/>
    <item code="9627Q"/>
  </deletion>
]
]
  <deletion type="brand">
    <brand code="13355011000036109" item-code="10269L"/>
    <brand code="926215011000036104" item-code="10269L"/>
    <brand code="13355011000036109" item-code="10270M"/>
    <brand code="926215011000036104" item-code="10270M"/>
    <brand code="13355011000036109" item-code="10296X"/>
```

# MPS Summary of Changes

```
</deletions>
<alterations>
  <alteration type="brand-name"/>
  <alteration type="form-strength"/>
  <alteration type="prescriber-group"/>
  <alteration type="manufacturer">
    <brand code="11414011000036100" item-code="2832C"/>
    <brand code="926143011000036105" item-code="5480K"/>
    <brand code="926141011000036109" item-code="6280M"/>
    <brand code="926144011000036107" item-code="8064K"/>
    <brand code="926140011000036102" item-code="8133C"/>
    <brand code="926143011000036105" item-code="8134D"/>
    <brand code="13944011000036101" item-code="8717T"/>
    <brand code="13945011000036102" item-code="8718W"/>
    <brand code="13946011000036109" item-code="8719X"/>
    <brand code="13947011000036107" item-code="8720Y"/>
    <brand code="76672011000036108" item-code="9183H"/>
    <brand code="926141011000036109" item-code="9568N"/>
  </alteration>
  <alteration type="restriction">
    <item code="10003L"/>
    <item code="10184B"/>
    <item code="10196P"/>
  </alteration>
</alterations>
```

# MPS Summary of Changes

```
</alterations>
<advance-notices>
  <advance-notice type="brand">
    <item code="8023G"/>
    <item code="8024H"/>
    <item code="9234B"/>
    <item code="9235C"/>
    <item code="10141R"/>
    <item code="10145Y"/>
    <item code="9485F"/>
    <item code="9486G"/>
  </advance-notice>
</advance-notices>
```

- **Data overlap**
  - Much of the source data looks the same.
    - It isn't easy to determine which values are correct or when you have used the wrong value.
  - Correct decisions may become wrong.

# Protection from volatility

- **MPS API**
  - Consequences of change are always limited to the distance between the source of the change and the API.

# Example: Summary of Changes

- **Addition Items**
  - Pull data by item code
    - Use service: `/0.3/items/{code}`
- **Addition Brands**
  - Pull data by brand code
    - Use service: `/0.3/brands/{code}`
- **Addition Notes**
  - Pull data by note code
  - Use service: `/0.3/notes/{code}`

# Example: Summary of Changes

- **Deletion Items**
  - Pull last month of data by item code
    - Use service: `/0.3/items/{code}?effectivedate=2015-07-01`
- **Deletion Brands**
  - Pull last month date by brand code
    - Use service: `/0.3/brands/{code}?effectivedate=2015-07-01`
- **Deletion Notes**
  - Pull last month data by note code
    - Use service: `/0.3/notes/{code}?effectivedate=2015-07-01`



# Example: Summary of Changes

- Alteration
  - Brand name
  - Form strength
  - Prescriber group
  - Manufacturer
  - Restriction
  - Authority type
  - Max quantity
  - Pack quantity
  - Number of repeat

# Example: Summary of Changes

- Alteration (continue)
  - Pull data by item code
    - Current month:  
`/0.3/items/{code}?effectivedate=2015-08-01`
    - Last month:  
`/0.3/items/{code}?effectivedate=2015-07-01`

# Example: Summary of Changes

- **Advance notice**

- Pull data by item code

- Current month:

- /0.3/items/{code}?effectivedate=2015-08-01

- Check the element

- AdvanceNotice.Type
- AdvanceNotice.EffectiveDate

<http://api.pbs.gov.au/0.3/items/8023G.xml?effectivedate=2015-08-01&advancenotice=true>

```
<AdvanceNotice.Type>deletion
</AdvanceNotice.Type>
```

```
<AdvanceNotice.EffectiveDate>2015-10-01
</AdvanceNotice.EffectiveDate>
```

- **Business Objectives**
  - Create a portable, lightweight representation of the data as an alternative to the text files.
    - Minimal dependencies.
    - Zero licensing issues.
  
- **The answer**
  - SQLite database.

# MPS SQLite

The screenshot shows the SQLite Manager application window. The title bar reads "SQLite Manager - D:\1\sqlite\PBS.sqlite". The menu bar includes Database, Table, Index, View, Trigger, Tools, and Help. The toolbar contains icons for refresh, copy, paste, save, and other functions. The left sidebar shows the database structure for "PBS.sqlite", including a "Master Table (1)" and "Tables (12)", with the "drug" table selected. The main window has tabs for Structure, Browse & Search, Execute SQL, and DB Settings. The "Execute SQL" tab is active, showing a text area with the query "SELECT \* FROM DRUG". Below the text area are buttons for "Run SQL" and "Actions", and a "Last Error" field showing "not an error". The main display area shows a table of results with columns: PROGR..., ATC\_C..., ATC\_T..., ATC\_P..., PBS\_C..., RESTR..., CAUTI..., NOTE..., MANN..., MAXIM..., and a sort icon. The table contains 15 rows of data. The status bar at the bottom shows "SQLite 3.8.9", "Gecko 40.0.2", "0.8.3.1-signed", "Exclusive", "Number of Rows Returned: 8608", and "ET: 642 ms".

PROGR...	ATC_C...	ATC_T...	ATC_P...	PBS_C...	RESTR...	CAUTI...	NOTE...	MANN...	MAXIM...	
CA	J05AF06	P	J05AF06	10356C	A	N	N	Oral	8	
CA	J05AF06	P	J05AF06	10294T	A	N	N	Oral	120	
CA	J05AR02	P	J05AR02	10357D	A	N	N	Oral	60	
CA	J05AR04	P	J05AR04	10305J	A	N	N	Oral	120	
GE	L04AA24	P	L04AA24	1220F	A	N	Y	Injection	4	
GE	L04AA24	P	L04AA24	1221G	A	N	Y	Injection	4	
HB	L04AA24	P	L04AA24	5605B	A	N	Y	Injection	1	
HS	L04AA24	P	L04AA24	9621J	A	N	Y	Injection	1	
GE	B01AC13	P	B01AC13	8048N	A	N	N	Injection	3	
GE	L02BX03	P	L02BX03	2698B	A	N	Y	Oral	120	
GE	N07BB03	P	N07BB03	8357W	A	N	Y	Oral	180	
GE	A10BF01	P	A10BF01	8189B	U	N	N	Oral	90	
GE	A10BF01	P	A10BF01	8188Y	U	N	N	Oral	90	

- A lightweight approach suited to modern web architectures.
  - JSON connects directly to the API.
  - Use Javascript to process in the browser.
  - Minimal learning required.
  - Very little infrastructure required.
  - More usable than the text files but based on modern technology.

## JSON Syntax

- Constructed from the XML syntax
  - Namespace is ignored
- Using the following rules:

XML	JSON
Attribute	JSON Value-pair
Element with no child element	JSON Value-pair
Element with children	JSON Object
Sequence of same elements	JSON Array

- XML

```
- <metadata version="0.3">  
  <effective-date>2015-08-01</effective-date>  
  <last-modified-date>2015-07-15T14:36:53+10:00</last-modified-date>  
  <data-version>0.3</data-version>  
  <library>3.0.21</library>  
</metadata>
```

- JSON

```
{  
  metadata: {  
    version: 0.3,  
    effective-date: "2015-08-01",  
    last-modified-date: "2015-07-15T14:36:53+10:00",  
    data-version: 0.3,  
    library: "3.0.21"  
  }  
}
```



- How mature is the API?
  - Used by the MPS since July 2011 opened to the public in August 2012.
    - Powers the general PBS Website.
    - Powers the mobile PBS Website.
  - Recently upgraded to 0.3 (4<sup>th</sup> version).
    - Now used for the hardcopy and Summary of Changes.
  - Undergoes extensive automated testing each month.
  - Is manually reviewed by a lot of people each month (community and industry).
  - Previous API versions are kept up-to-date until all users have migrated forward.

Where is it?

<http://api.pbs.gov.au>

## How to access it?

- Sign up by sending an email to:  
[pbs.websmaster@health.gov.au](mailto:pbs.websmaster@health.gov.au)
- You will be given a unique API Key
  - To give you access to the API
  - To protect the system from abuse
  - To allow us to optimise the data based on real usage

## What formats are supported?

- The API can deliver the PBS data
  - as **XML**
  - Or as a **JSON** object
- Also supports **JSONP**

## More information

- API Documentation

<https://dev.pbs.gov.au/docs/>

- To familiarise yourself with API

<http://api.pbs.gov.au/howto.html>

- Developer online forum

<https://dev.pbs.gov.au/discussions.html>