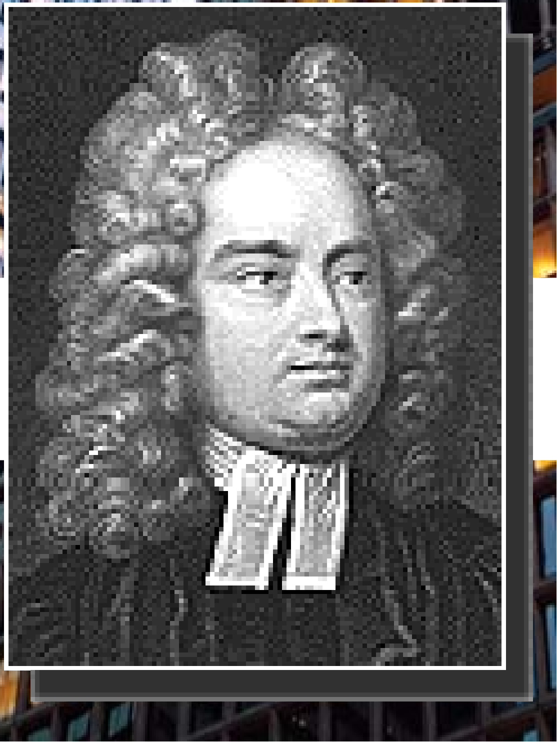
A Modest Proposal

For Preventing Middleware Research From Being a Burden to Industry, and For Making It Beneficial to The Public

john (jonathan) swift wilkes

Middleware panel 2006



Problem: Middleware is under threat

- industry wants “relevance”
- the academic community wants “rigor”
- it’s getting too easy for others to join in

- root cause:
  - Middleware has become **too simplistic**

What to do about it?

- **complexity is good for Middleware**

- employment for you + your students
- endless chances to innovate
- keeps others at bay
  - less risk of “we did that a decade ago”

What to do about it?

Resist misguided calls for "simplicity"

- Most problems between approaches result from **over-simplification**
  - your approach doesn't handle this corner case
  - 100% solutions are essential
  - every deployment has unique requirements

What to do about it?

Focus on mechanisms, not goals

yes: simple-looking **interfaces**

- can readily be measured (e.g., lines of code, number of methods)

no: clear **goals and metrics**

- this requires careful thought
- eliminates many avenues for complexity

What to do about it?

ruthlessly eliminate alternatives

Example: **(a)synchronous invocations**:

- it would be a disaster if people realized just how easy both techniques were to understand

- insist on just <u>one</u> choice
  - lots of openings for baroque workarounds

What to do about it?

Premature standardization is great!

- fossilizes poor ways of achieving end goals
  - more opportunities for exotic workarounds

- lack of consensus ➔ include <u>all</u> alternatives

- doesn’t require proof of usefulness

What to do about it?

Emphasize the irrelevance of customer needs

- today’s customers are stuck on yesterday’s problems
  - ➔ their guidance can safely be ignored

What to do about it?

Pooh-pooh “interoperability”

- it’s just an excuse for somebody to impose their answer on you

- it requires thorough understanding of an alternative point of view

What to do about it?

Obfuscatory representations

only wimps can’t think in raw XML!

- example: why say

```
{diskDrive:u
  {serialNumber "1234-5678"}
}
```

- when you could have said:

```
<sst:object type="diskDrive"
name="u"> <sst:object type="serialNumber">
<cbt:string>1234-5678</cbt:string> </sst:object>
</sst:object>
```

What to do about it?

Obfuscatory representations

which better hides the intent from outsiders?

```
#>Latin-2-1 Rome-2-3
{objectType:nougat {
    {objectType:serialNumber {isA string}}
    {objectType:targets      {isA string_list}}
    {objectType:edible       {isA object}
                              {objectType:tasty  {isA bool}}
                              {objectType:poison {isA bool}}
    }
    {objectType:comment      {isA string}}
    {objectType:cost         {isA dm}}
}
{nougat:squirrel_3 {
    {serialNumber "1234-5678"}
    {targets (tooth_1 tooth_2) }
    {edible:reality {tasty false}{poison false}}
    {edible:gourmet {tasty true}}
    {comment "A funny string, this,
with an &amp; (ampersand) in it."}
    {cost {dm_normal {mean 123.45}{stddev 6.78}}
}
```

not XML

```
•<?xml version="1.0" ?>
•<!DOCTYPE ROME SYSTEM "http://cello.hpl.hp.com/ssp/Rome/Rome2/Greek-
1.0.dtd">
•<Rome RomeVersion="2.2" DTDversion="1.0">

•<object type="objectType" name="nougat">
•    <object type="objectType" name="serialNumber">
•        <object type="isA"><str>string</str></object>
•    </object>
•    <object type="objectType" name="targets">
•        <object type="isA"><str>string_list</str></object>
•    </object>
•    <object type="objectType" name="edible">
•        <object type="isA"><str>bool</str></object>
•        <object type="objectType" name="tasty">
•            <object type="isA"><str>bool</str></object>
•        </object>
         <object type="objectType" name="poison">
•            <object type="isA"><str>bool</str></object>
•        </object>
•    </object>
•    <object type="objectType" name="comment">
•        <object type="isA"><str>comment</str></object>
•    </object>
•    <object type="objectType" name="cost">
•        <object type="isA"><str>dm</str></object>
•    </object>
•</object>

•<object name="squirrel_3" type="nougat">
•    <object type="serialNumber"><str>1234-5678</str></object>
•    <object type="targets"><list>
•        <name>tooth_1</name>
•        <name>tooth_2</name>
•    </list></object>
•    <object type="edible" name="reality">
•        <object type="tasty"><bool>true</bool></object>
•        <object type="poison"><bool>false</bool></object>
•    </object>
•    <object type="edible" name="gourmet">
•        <object type="tasty"><bool>false</bool></object>
•    </object>
•    <object type="comment"><str>A funny string, this,
•with an &amp; (ampersand) in it.</str></object>
•    <object type="cost">
•        <object type="mean">  <num>123.45</num></object>
•        <object type="stddev"><num>6.78  </num></object>
•    </object>
•</object>

•</Rome>
```

XML

help save Middleware!

join the new
ACM **Campaign for Actively Complex Middleware (CACM)**