

Access Management in AIMMS WebUI apps

February 22, 2017

Chris Kuip

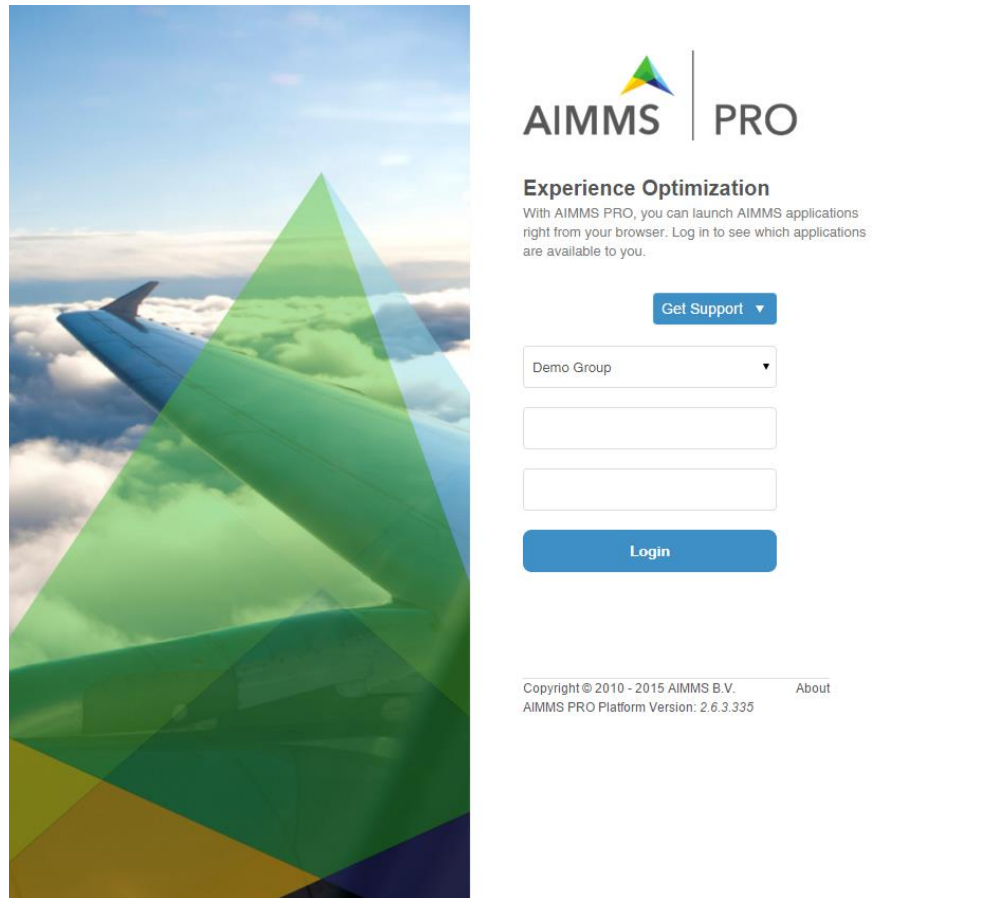


Agenda

1. AIMMS PRO Overview
2. AIMMS PRO Publishing and Using
3. Controlling read and write access to data in an app
4. Data management via WebUI
5. WebUI Widgets for file uploading and downloading
6. File storing in AIMMS PRO storage

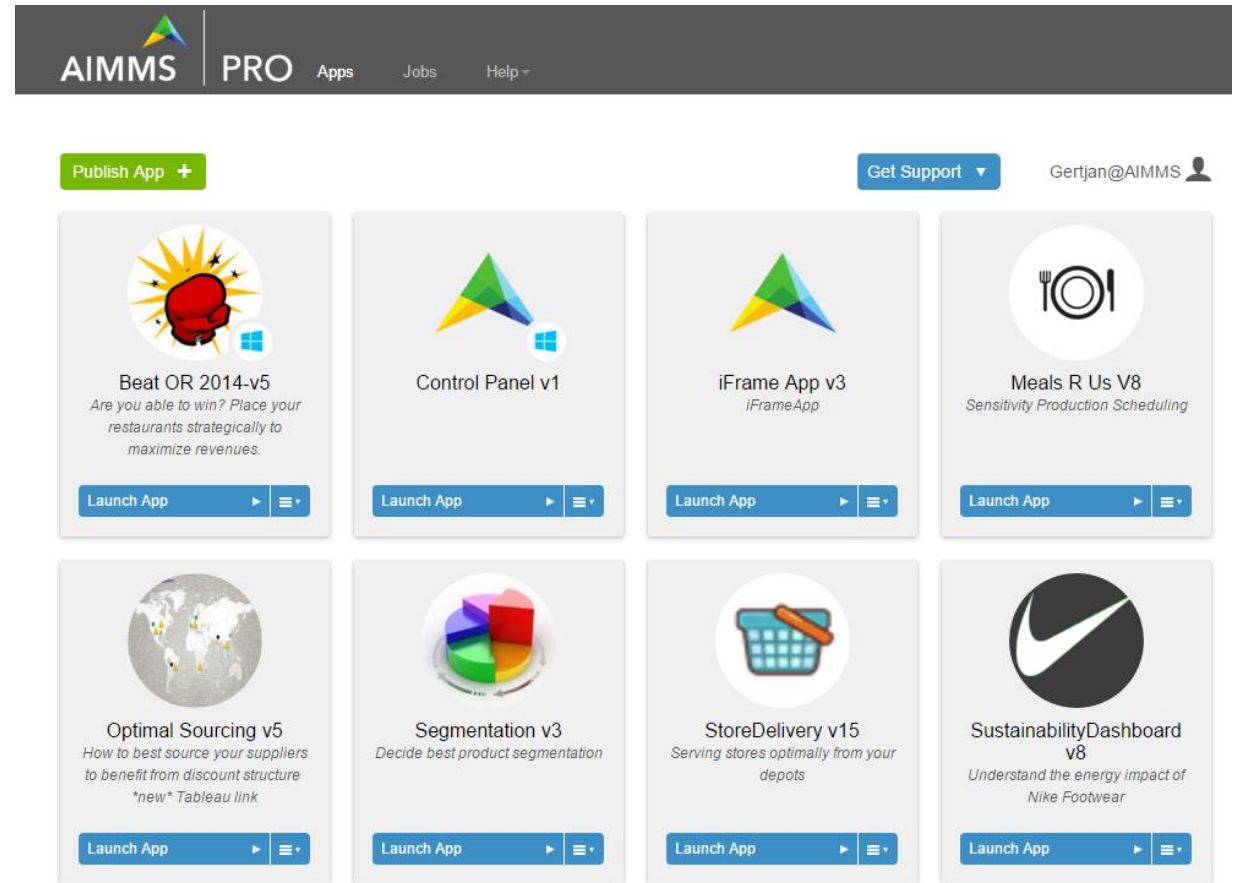
1a. AIMMS PRO: Your Personal App store

Logon



The logon page features a large background image of an airplane flying over a green, geometric landscape. The AIMMS PRO logo is at the top left. Below it, the text 'Experience Optimization' is followed by a short description: 'With AIMMS PRO, you can launch AIMMS applications right from your browser. Log in to see which applications are available to you.' There is a 'Get Support' button, a dropdown menu for 'Demo Group', two empty input fields, and a 'Login' button. At the bottom, there is a copyright notice: 'Copyright © 2010 - 2015 AIMMS B.V. AIMMS PRO Platform Version: 2.6.3.335' and an 'About' link.

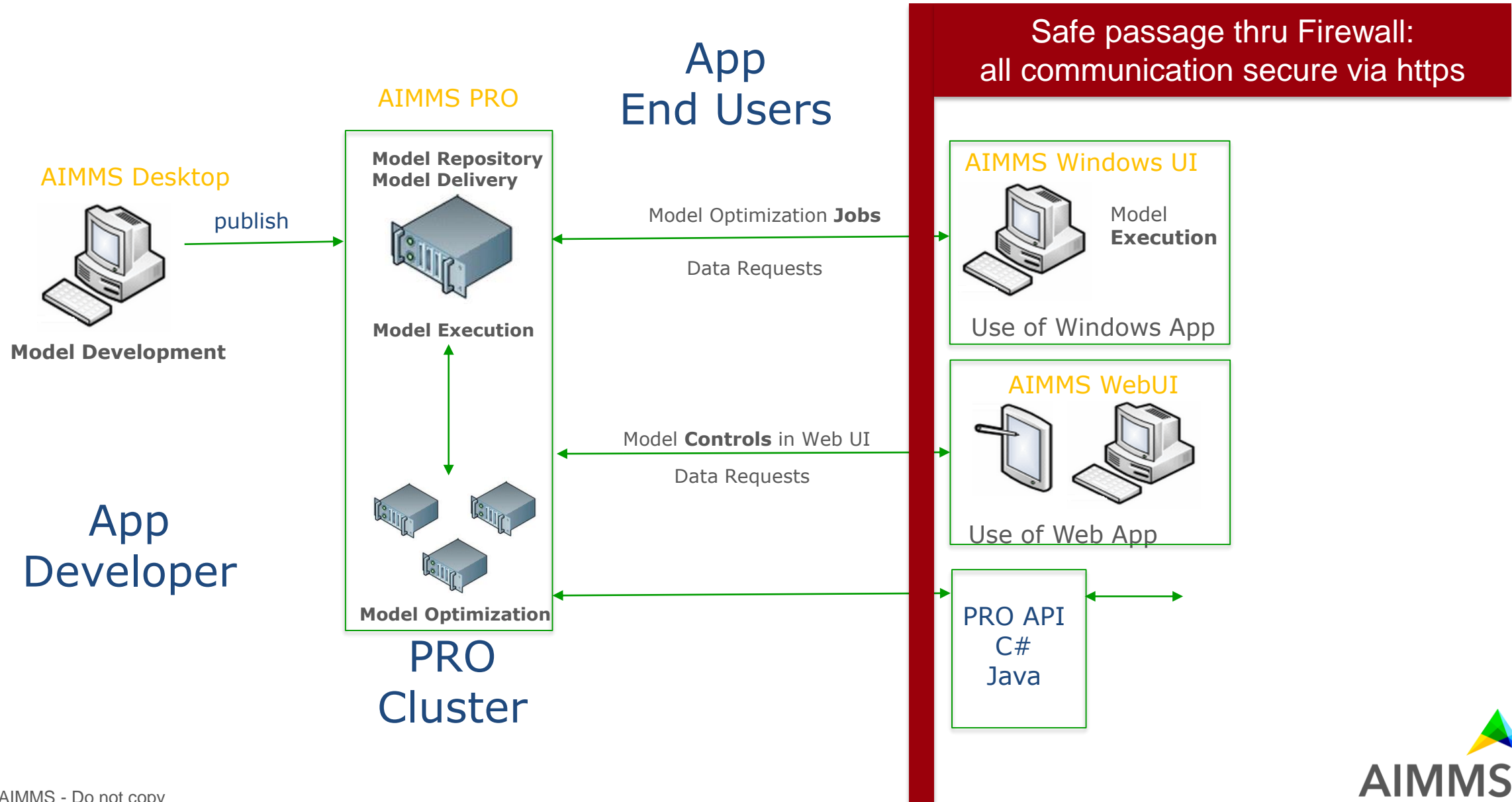
Menu of apps



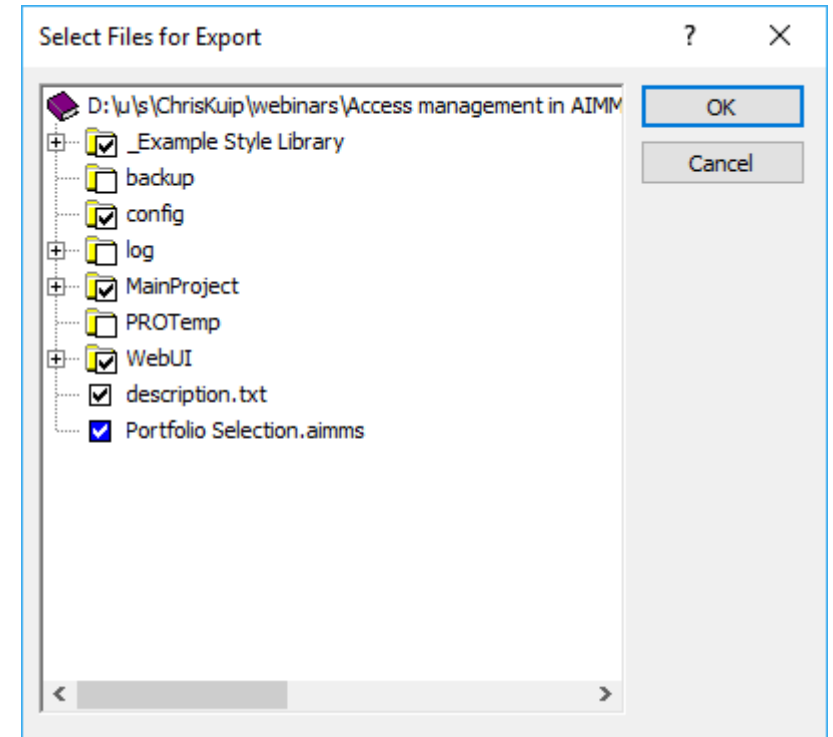
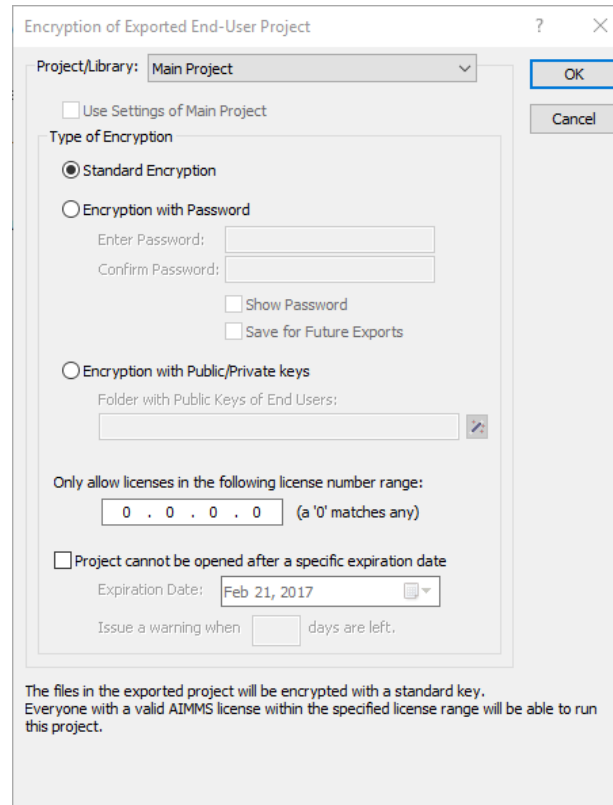
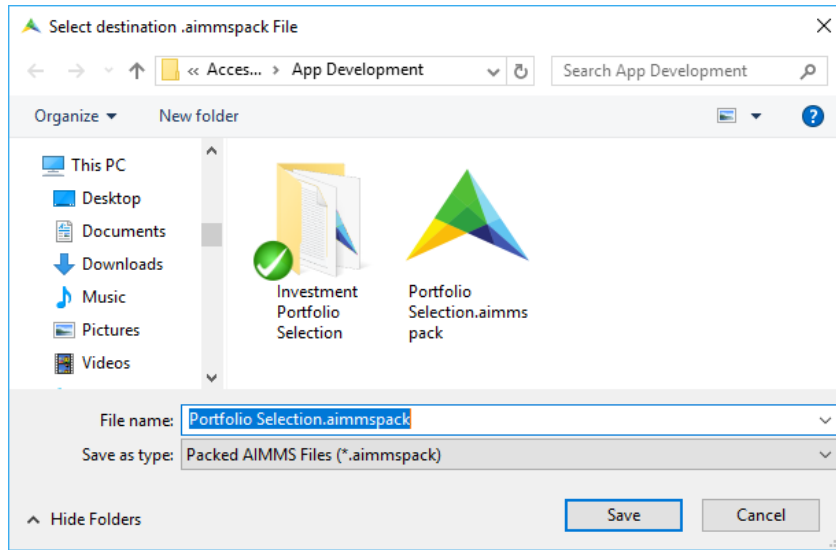
The menu of apps page has a dark header with the AIMMS PRO logo and navigation links for 'Apps', 'Jobs', and 'Help'. A 'Publish App +' button is on the left, and 'Get Support' and a user profile 'Gertjan@AIMMS' are on the right. The main area displays eight app cards in a 2x4 grid. Each card includes an icon, the app name, a brief description, and a 'Launch App' button with a menu icon.

App Name	Description
Beat OR 2014-v5	Are you able to win? Place your restaurants strategically to maximize revenues.
Control Panel v1	
iFrame App v3	iFrameApp
Meals R Us V8	Sensitivity Production Scheduling
Optimal Sourcing v5	How to best source your suppliers to benefit from discount structure *new* Tableau link
Segmentation v3	Decide best product segmentation
StoreDelivery v15	Serving stores optimally from your depots
SustainabilityDashboard v8	Understand the energy impact of Nike Footwear

1b. Typical AIMMS PRO 2.x Architecture with AIMMS Apps



2a. AIMMS PRO publishing – Creating an .aimmspack file




2b Setting the permissions of the application





PortfolioSelection 8
Permission settings


Change History


Clear



AIMMS
(AD)
● r ● w ● x


Procurement
(AD)
● r ● w ● x


JoeB
@AIMMS
(AD)
● r ● w ● x


ROOT
● r ● w ● x


Managers
(AD)
● r ● w ● x

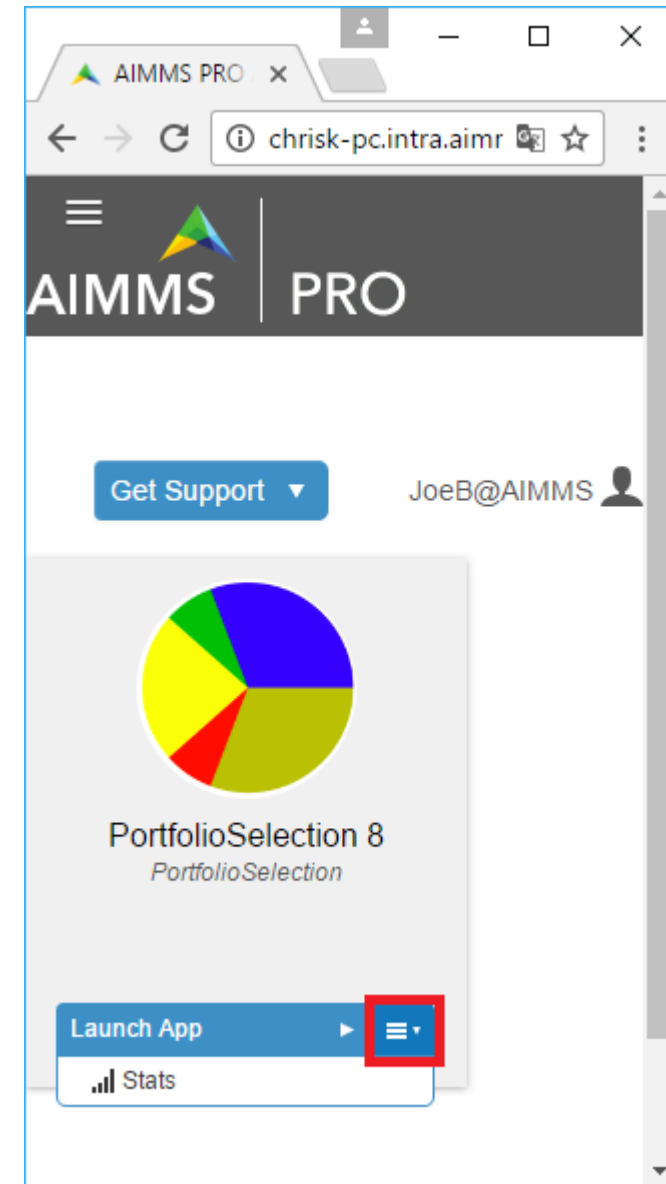
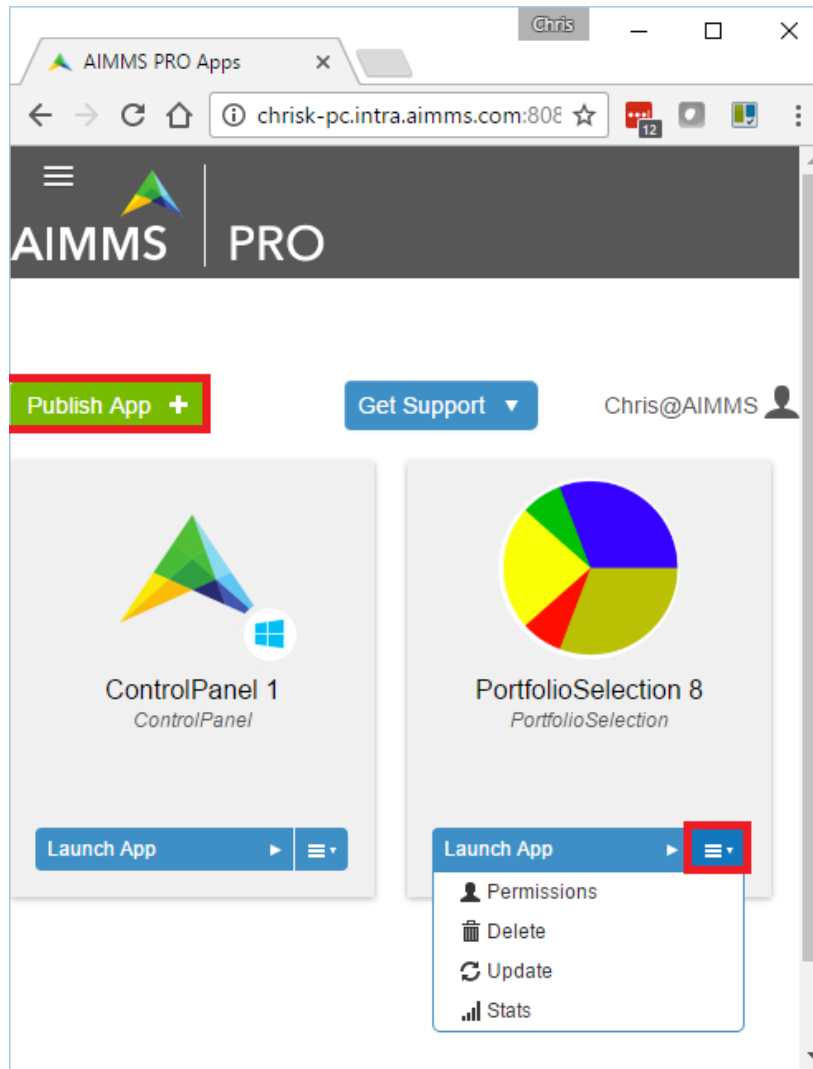

Domain Users
(AD)
● r ● w ● x

Active Directory Domain Roles
Correspond to
AIMMS PRO groups

Get Support

Chris@AIMMS

2b. Pro Portal View – Developer and End User



3. Using the APP by users from different groups

No write access

Covariances			
	Stocks	Bonds	Real estate
Stocks	2,25	-0,12	0,45
Bonds	-0,12	0,64	0,34
Real estate	0,45	0,34	1,44

With write access

Covariances			
	Stocks	Bonds	Real estate
Stocks	2,25	-0,12	0,45
Bonds	-0,12	0,64	0,34
Real estate	0,45	0,34	1,44

- > Modifiability: black indicates not modifiable, blue does.
- > Access at all: visibility of data and widgets can be controlled.
- > How can we control this from within the application?

3a. In app user identification

```
! Identify the current app user and the groups to which this user belongs.
if pro::GetPROEndPoint() then
    pro::GetCurrentUserInfo(pro_cur_env, pro_cur_user,
        |         pro_cur_Bucket, pro_usr_groups, pro_cur_Group_Name);
else
    ! Running as Application Developer.
    pro_cur_env      := "AIMMS" ;    ! Scalar string parameter
    pro_cur_user     := "Chris" ;    ! as well
    pro_cur_Bucket   := "PROTemp" ; ! as well
    pro_usr_groups   := data {};     ! Collection of entities, mostly the groups
    empty pro_cur_Group_Name ;      ! Group Descriptions, indexed string parameter
endif ;
```

> Pro::GetCurrentUserInfo (...)

Identifies the current user and his/her roles.

> pro::GetPROEndPoint () Returns the URL of the hosting AIMMS PRO system.

When empty, we're running via AIMMS developer environment.

3b. Using User Info to assign visibility and modifiability rights.

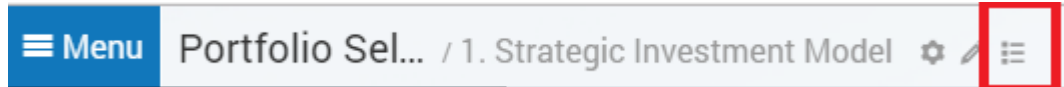
> Initialize

```
! AllPublicIdentifiers defines the identifier visible in the model  
! Typically, the declarations inside the main model:  
AllPublicIdentifiers := Main_Portfolio_Selection ;  
  
! As a precaution, make AllPublicIdentifiers read only  
CurrentInputs -= data { AllPublicIdentifiers,  
                        | ! Make the covariance data read only  
                        | covariance } ;
```

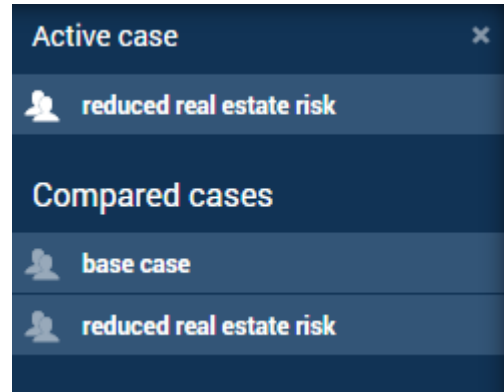
> Use User Identification

```
! The current APP user is a member of the AIMMS PRO user group Procurement or:  
if exists( pug | pro_cur_Group_Name( pug ) = "Procurement" ) or  
  ! The app developer  
  ( pro_cur_user = "Chris" and pro_cur_env = "AIMMS" )  
then  
  CurrentInputs += ! Make covariance data modifiable.  
  | data { covariance } ;  
endif ;
```

4 Scenario Data



> Activate Data Manager



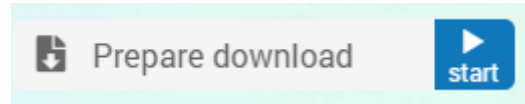
> Share and select Cases

Covariances						
	base case	base case	base case	reduced rea	reduced re	reduced rea
	Stocks	Bonds	Real estate	Stocks	Bonds	Real estate
Stocks	2,25	-0,12	0,45	2,25	-0,12	0,00
Bonds	-0,12	0,64	0,34	-0,12	0,64	0,00
Real estate	0,45	0,34	1,44	0,00	0,00	1,44

> Covariance Overview

5a Download button

> Download button:

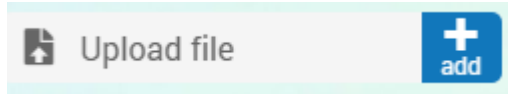


> Download Procedure:

Procedure	DownloadCovariance
Arguments	(FileLocation, StatusCode, StatusDescription)
Property	
Body	<pre>Block FileLocation := "covar.put"; SP_ReportName := webui::GetIOFilePath(FileLocation); DownloadCovarianceActuallyCreateFile(); StatusCode := webui::ReturnStatusCode('CREATED'); StatusDescription := "Nice."; OnError err Do StatusCode := webui::ReturnStatusCode('ERROR'); StatusDescription := errh::Message(err); errh::MarkAsHandled(err); EndBlock ;</pre>

5b Upload Button

> Upload Button:



> Upload Procedure:

Procedure	UploadProc
Arguments	(FileLocation, StatusCode, StatusDescription)
Property	
Body	<pre>Block sp_FileUploadLocation := webui::GetIOFilePath(FileLocation); read from file sp_FileUploadLocation ; FileDelete(sp_FileUploadLocation); StatusCode := webui::ReturnStatusCode('OK'); StatusDescription := "All perfect!"; OnError err Do StatusCode := webui::ReturnStatusCode('ERROR'); StatusDescription := errh::Message(err) ; errh::MarkAsHandled(err); EndBlock ;</pre>

6a. Personal File Storing

> Save Button

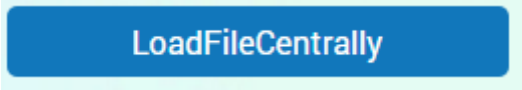
saveFileCentrally

> Save Procedure

Procedure	SaveCovarFileCentrally
Arguments	
Property	
Body	<pre>sp_ErrorMessage := "" ; Block write Covariance to file "covar.put" ; Pro::SaveFileToCentralStorage("covar.put", "pro:/publicdata/portfolioselection/covar.put"); OnError err Do sp_ErrorMessage := errh::Message(err); errh::MarkAsHandled(err); Endblock ;</pre>

6B File Loading

> LoadFileButton



> LoadFileProc

Procedure	LoadCovarFileCentrally
Arguments	
Property	
Body	<pre>sp_ErrorMessage := "" ; Block Pro::RetrieveFileFromCentralStorage("pro:/publicdata/portfolioselection/covar.put", "covar.put"); read from file "covar.put" ; OnError err Do sp_ErrorMessage := errh::Message(err); errh::MarkAsHandled(err); Endblock ;</pre>

6C File sharing

> Sharing files:

Procedure	ShareCovarFileCentrally
Arguments	
Property	
Body	<pre>sp_ErrorMessage := "" ; Block write Covariance to file "covar.put" ; ep_ent_Procurement := first(guipro::PRO_ENT guipro::PROGroupName(guipro::PRO_ENT) = "Procurement"); ep_ent_Managers := first(guipro::PRO_ENT guipro::PROGroupName(guipro::PRO_ENT) = "Managers"); sp_authorization := formatString("#6+%e#4+%e", ep_ent_Procurement, ep_ent_Managers); Pro::SaveFileToCentralStorage("covar.put", "pro:/publicdata/portfolioselection/covar.put", sp_authorization); OnError err Do sp_ErrorMessage := errh::Message(err); errh::MarkAsHandled(err); Endblock ;</pre>

Recap

AIMMS WebUI applications published on AIMMS PRO have the following properties:

1. Access to these applications can be controlled at the group and individual level
2. Access to data can be controlled at the group and individual level
3. Cases can be saved, compared and shared
4. Files can be uploaded/downloaded and shared via the AIMMS PRO Storage space

Next Webinar

- > Marcel Hunting will present the next webinar on March 22, 2017.
- > The Topic will be: “A new multistart algorithm”

Questions