



Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Federal Office of Topography swisstopo

Automatic testing of Geodata and what's new in ProSuite

wissen wohin
savoir où
sapere dove
knowing where

December, 1st 2022

Federal Office of Topography swisstopo:
Stefan Flury



Agenda

- ProSuite a look back at history until today
- ProSuite scope
- ProSuite system architecture
- ProSuite PythonAPI
- LiveDemo



A look back at history until today

- Esri Switzerland developed ProSuite together with swisstopo for ArcMap. End of live 2026 is related to ArcMap
- Dira GeoSystems AG (Founded 2021) has taken over the source code
They port the code to ArcGIS Pro and ArcGIS Enterprise
They provide maintenance and support via subscription
the source code is mostly open source
<https://dirageosystems.ch/#home>



Scope

- ArcGIS Pro Add-in
 - Edit Tools
 - Quality Verification
 - Work Lists
- Administrations-Client
 - Client for configuration database
 - Projects, data models, quality rules
- Python API
 - Base for geoprocessing tools
 - Additional configuration variant for scripting
 - Call from any client



Scope for this presentation and demo

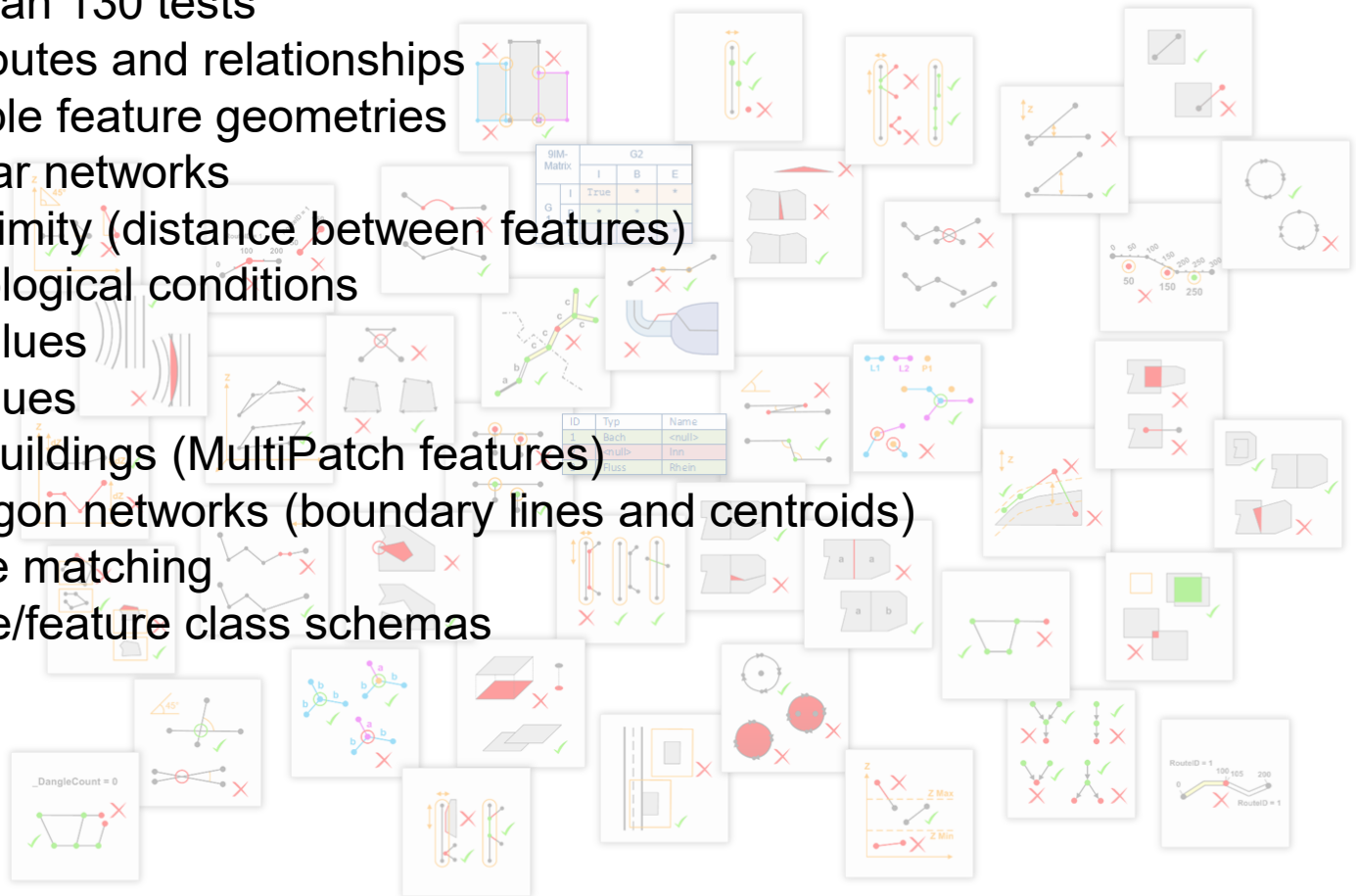
- ArcGIS Pro Add-in
 - Edit Tools
 - **Quality Verification**
 - Work Lists
- Administrations-Client
 - Client for configuration database
 - Projects, data models, quality rules
- Python API
 - **Base for geoprocessing tools**
 - Additional configuration variant for scripting
 - **Call from any client**



QA Conditions

More than 130 tests

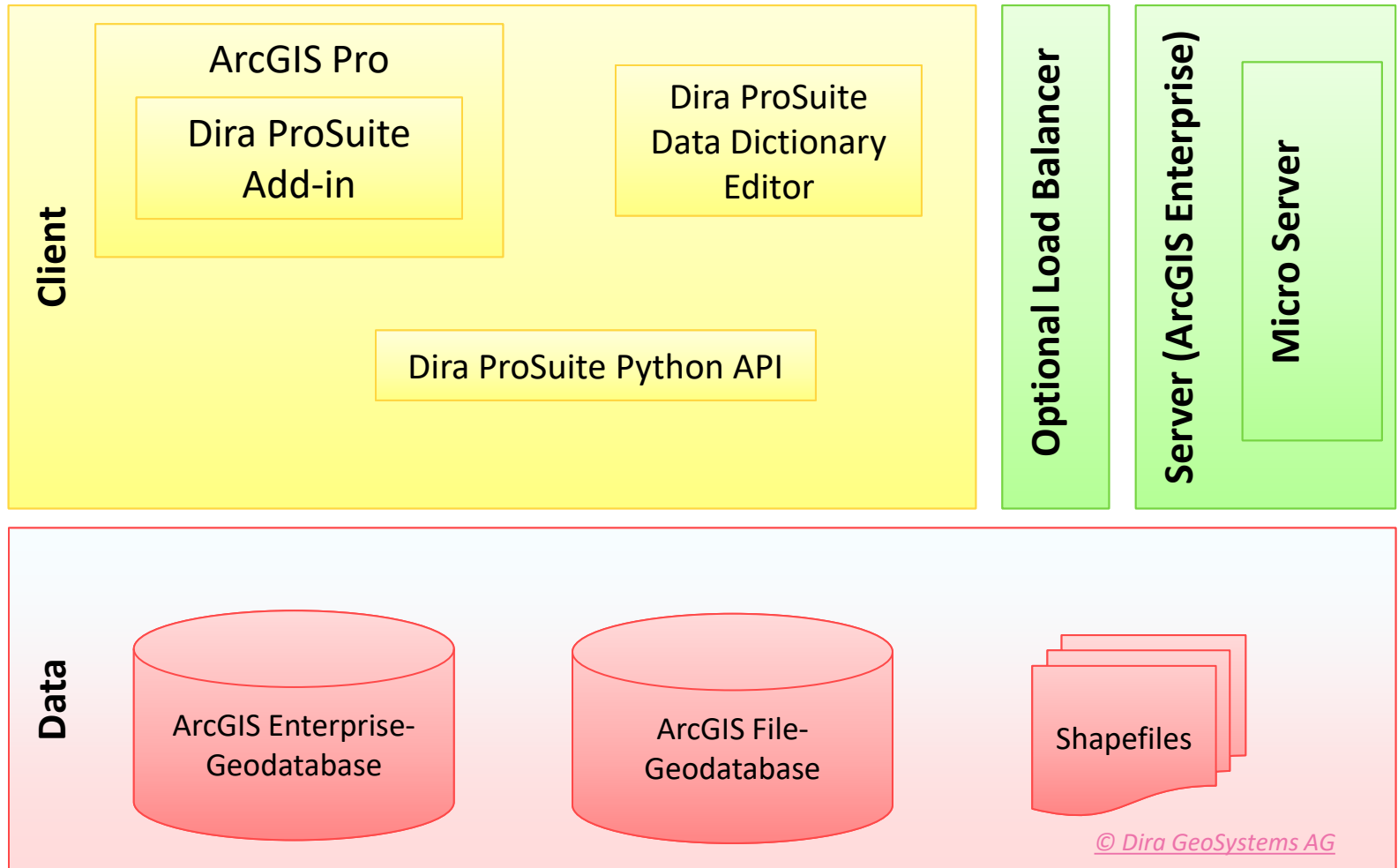
- Attributes and relationships
- Simple feature geometries
- Linear networks
- Proximity (distance between features)
- Topological conditions
- M values
- Z values
- 3D buildings (MultiPatch features)
- Polygon networks (boundary lines and centroids)
- Edge matching
- Table/feature class schemas
-



© 2014 Esri Schweiz AG

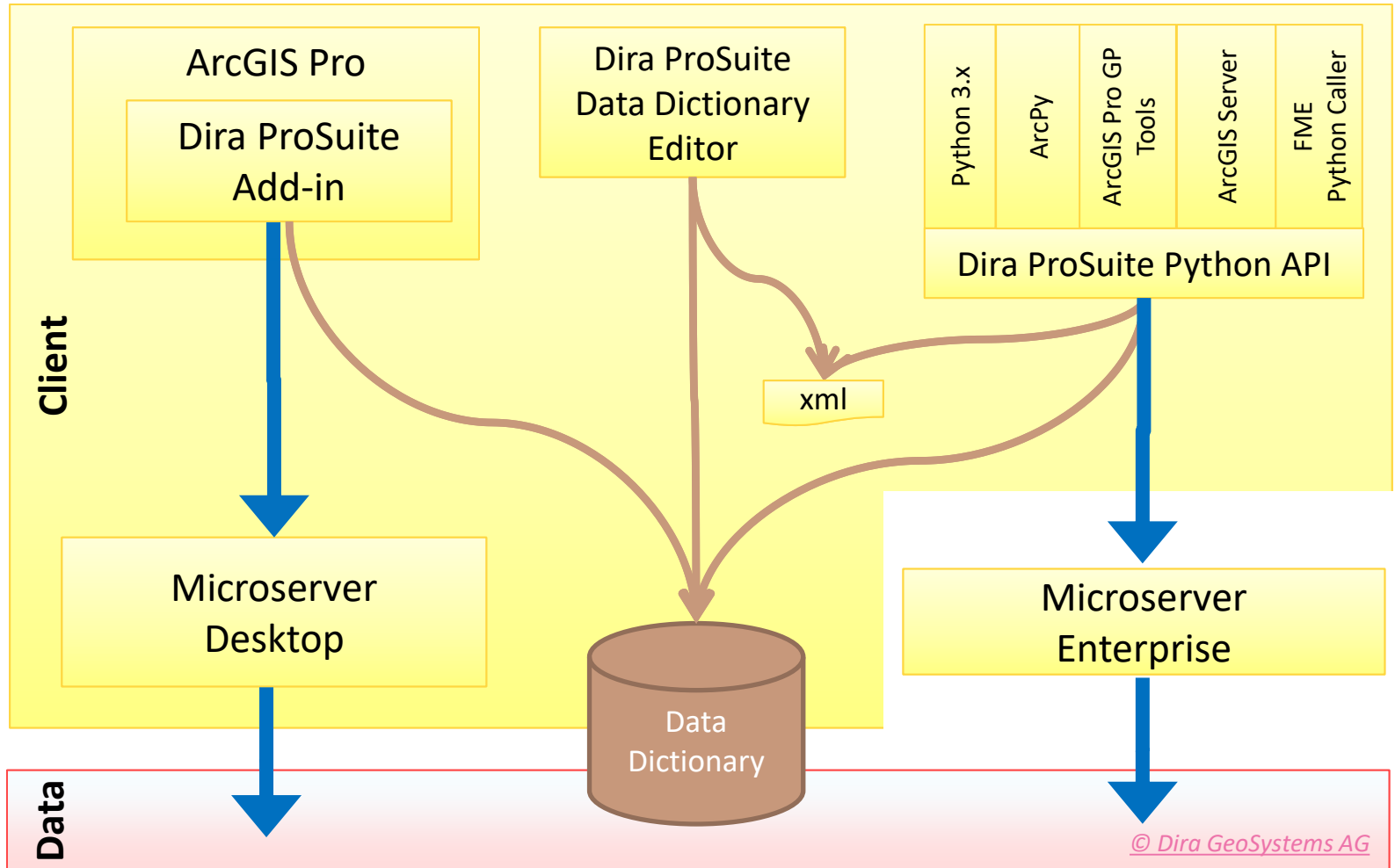


ProSuite system components



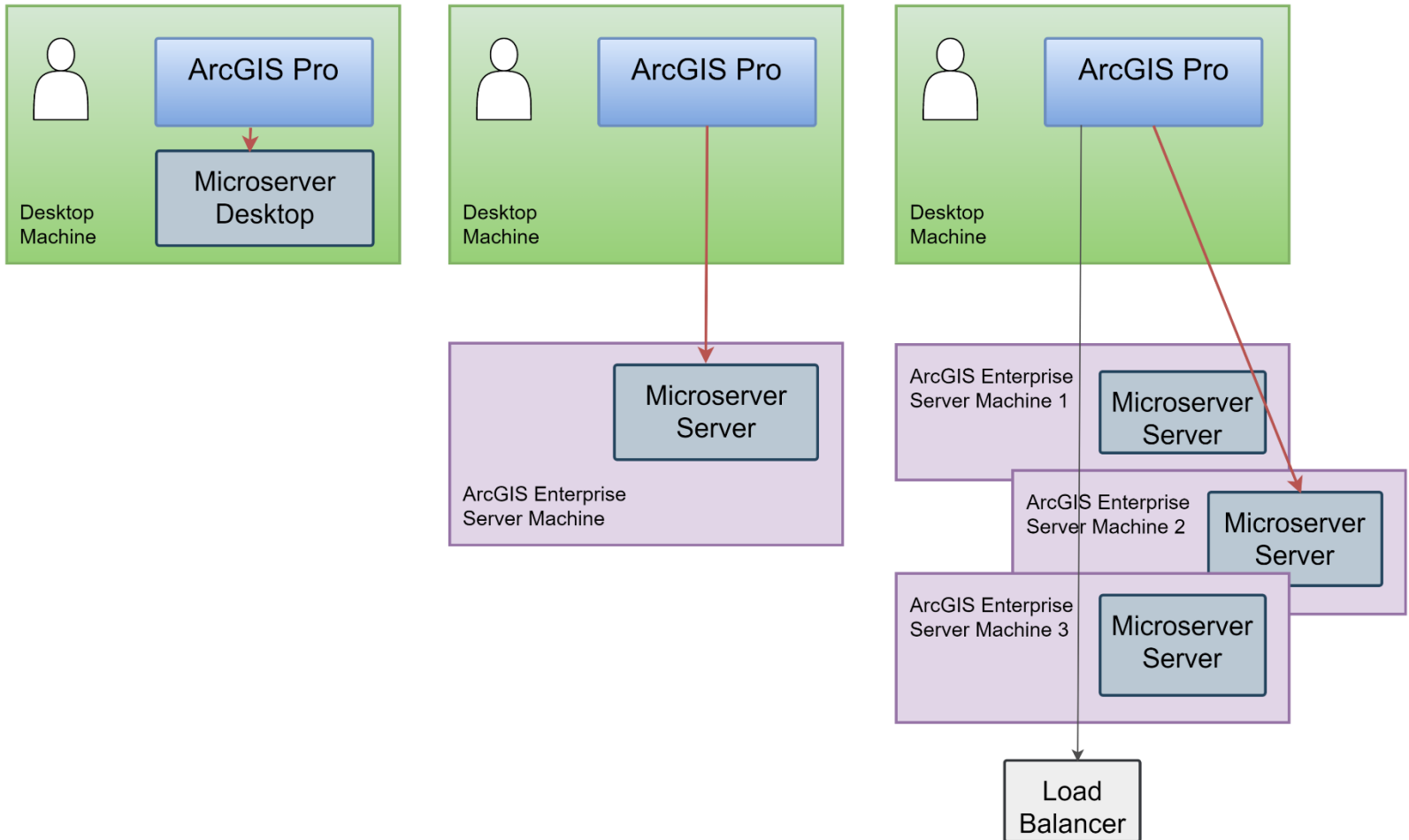


ProSuite system components Client





ProSuite Deployment Scenarios



© Dira GeoSystems AG



ProSuite PythonAPI Verification using XmlSpecification

1. create a Verification instance
2. define the Verification specification: create a XmlSpecification instance
3. define the Verification output directory
4. execute the verification

```
from prosuite.verification import Verification
from prosuite.xml_specification import XmlSpecification

verification = Verification(host_name='localhost', port_nr=5151)
verification.specification = XmlSpecification(
    specification_file='C:/temp/road_specification.qa.xml',
    specification_name="TLM_Produktionsunterstuetzung",
    data_source_replacements=[["TLM_Production", sde_file]])
verification.output_dir = 'C:/temp/verification_output'

for verification_response in verification.execute():
    print(verification_response.message_level)
    print(verification_response.service_call_status)
    print(verification_response.message)
```



Demo

1. Python script based on the documented sample
2. Python script based on our qa module. The module encapsulates some often used functions like:
 - Initialise the QA (determine where the microserver or load balancer is running)
 - Run a QA and get all messages back
 -
3. Python script based on a swisstopo project (simplified for this demo). The script can be used on the command line or as a GPTool.
4. GPTool



Questions

My question:

- May I pass on the list of participants to Dira?

Do you have questions?