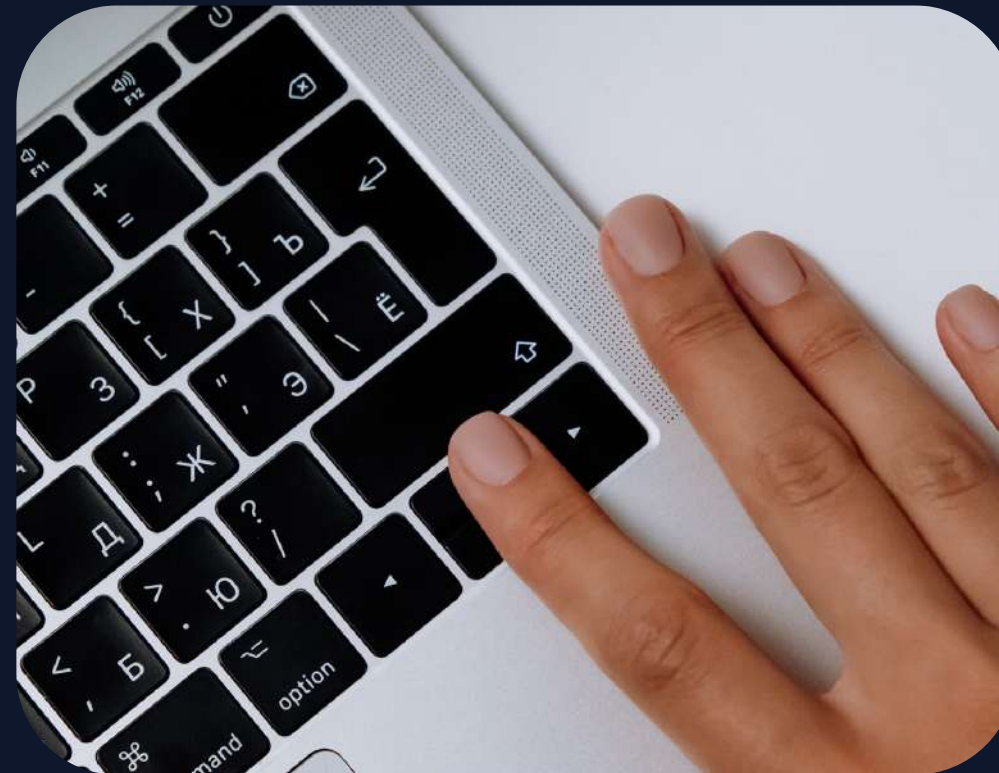
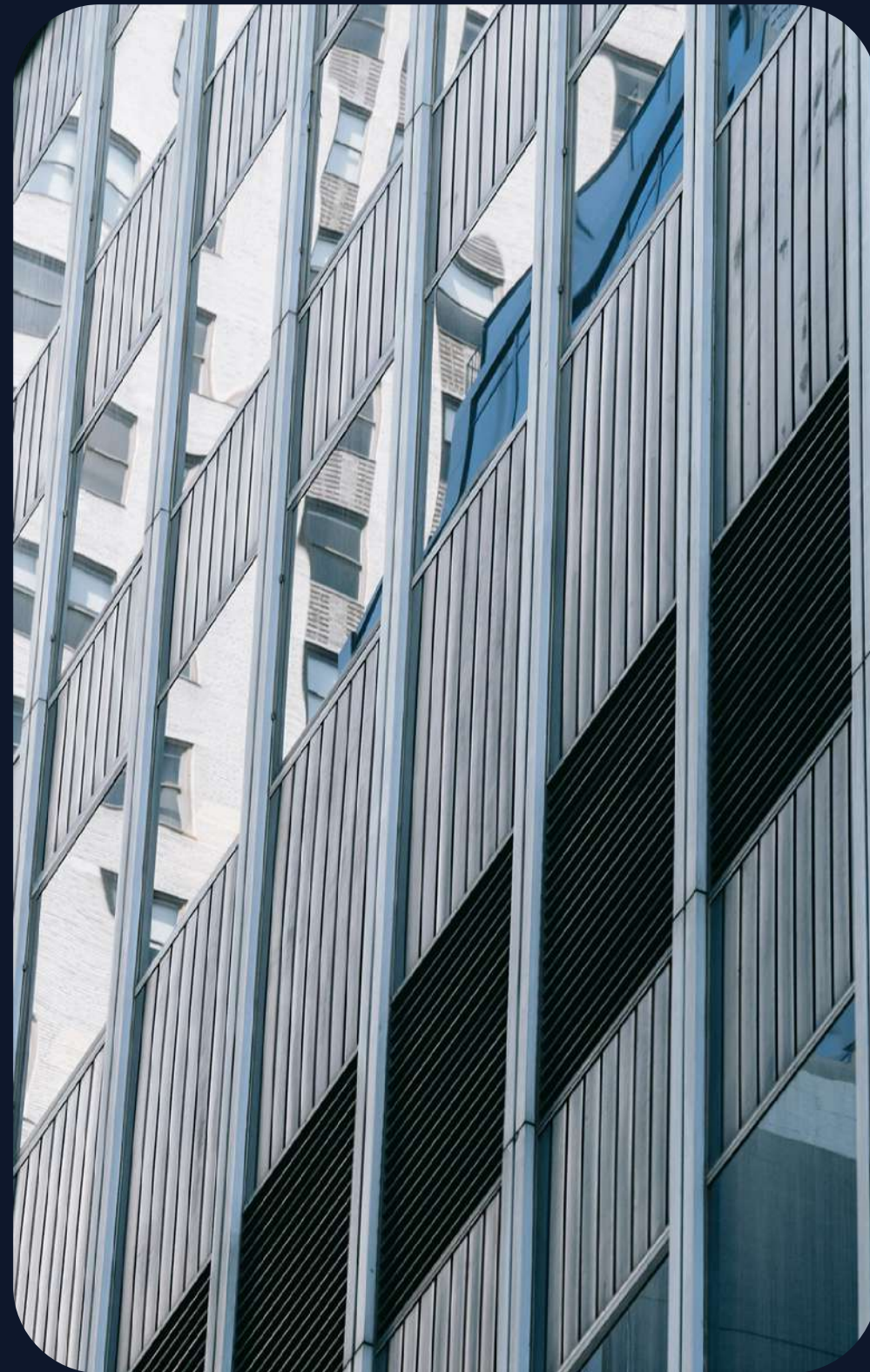


**MakeDoc  
for TIBCO**



  
**Behaim ITS**





# SERVICES PROVIDED

Each MakeDoc edition is available through an annual subscription.

Your subscription includes a license key and premium support services, covering:

- Assistance with installation and configuration
- Consultations for custom QA rule development
- Access to premium updates and priority bug fixes

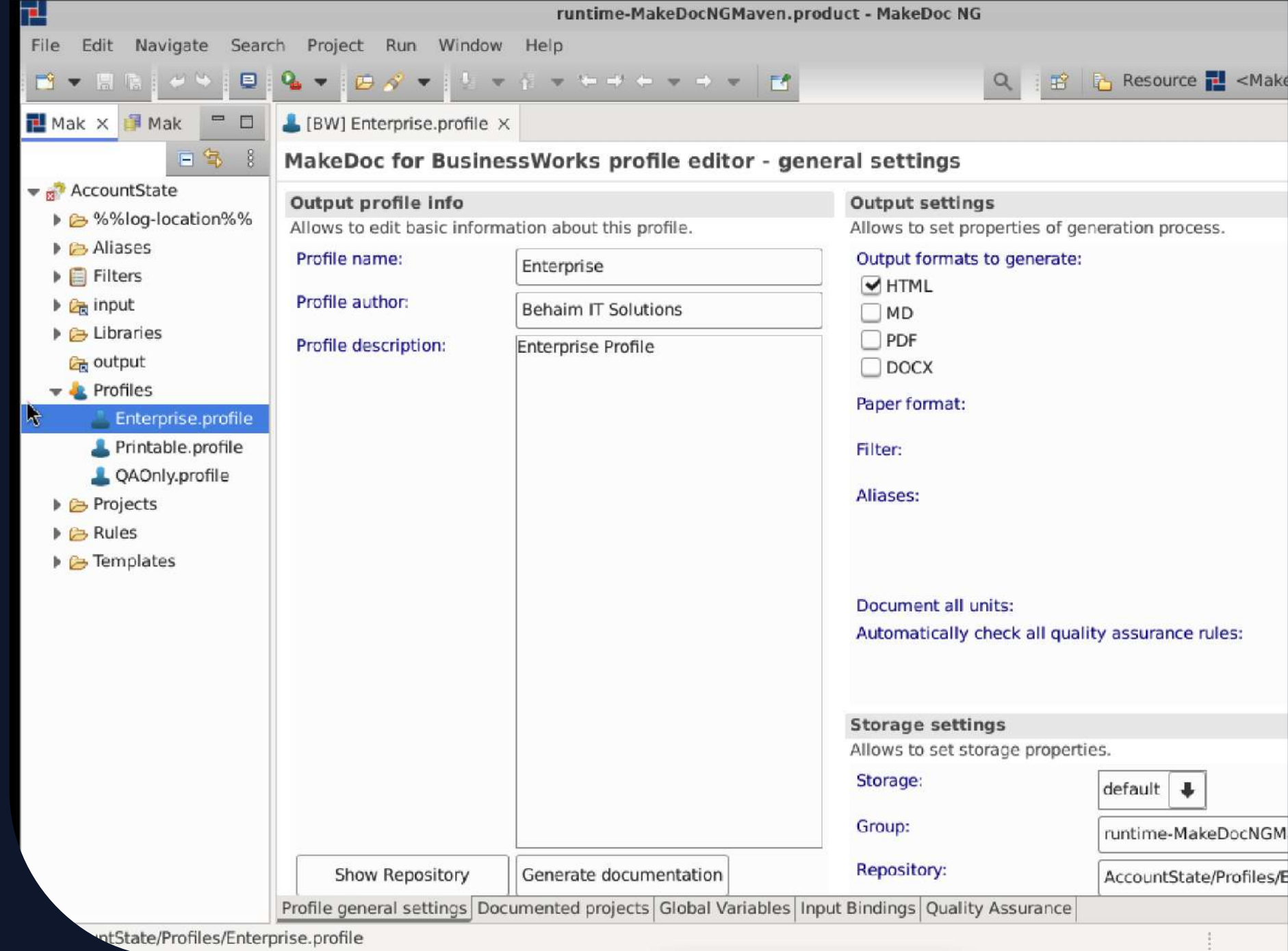
Renewing your subscription ensures uninterrupted access to the latest features, improvements, and expert support throughout the year.



# Developer

Desktop application – the best match for an individual use

MakeDoc Developer Edition is an Eclipse-based desktop application available for Windows, Linux, and macOS. It offers a wide range of configuration options, enabling you to generate documentation on demand that perfectly matches your current needs.



Create custom QA rules or produce professional documentation in MS Word, PDF, HTML, or Markdown – whenever you need it. Share your results effortlessly: print or email PDFs, edit DOCX files, or take your documentation to meetings with ease.

# Container

Natural choice for your cloud environments.

Just pull a Docker image, provide source and easily generate documentation and analysis without any special setup.

Integrate the run and die pattern into your pipelines and workflows and automate code review or simply push the documentation where it helps.

```
ARG MAKEDOC_VERSION=4.4.3
ARG MAKEDOC_ARCHIVE=MakeDoc${MAKEDOC_VERSION}
ARG MAKEDOC_SERVER_VERSION=4.4.3
ARG MAKEDOC_SERVER_ARCHIVE=makedoc_server-${M
ARG ANT_VERSION=1.10.13
# In kubernetes, this port must be different
# defaults to 8080
# On run, port can be remapped with --httpPor
# See https://www.jenkins.io/doc/book/install
ARG http_port=8080

ENV MAKEDOC_ROOT=/opt/makedoc
ENV JAVA_OPTS="-Djenkins.install.runSetupWiza

ENV JDK_JAVA_OPTIONS="--add-opens java.base/j
--add-opens java.base/java.util=ALL-UNNAMED \
--add-opens java.base/java.io=ALL-UNNAMED \
--add-opens java.base/java.security=ALL-UNNAM
--add-opens java.base/java.text=ALL-UNNAMED"

USER root

# CA certificates
#COPY certs/* /etc/pki/ca-trust/source/anchor
#RUN /usr/bin/update-ca-trust
```

# Server

Complex solution for your on-premise or virtualized environment

Automate documentation and code analysis using built-in Jenkins jobs or pipelines. Think smart and share documentation with all they need it. Make the troubleshooting or impact analysis easier than ever. Adjust the setup exactly matching your needs. Publish generated artefacts in your intranet

The screenshot shows the Jenkins dashboard for 'MakeDoc continuous documentation server'. It includes navigation links for 'New Job', 'People', 'Build History', and 'Manage Jenkins'. The 'Build Queue' section indicates 'No builds in the queue.' The 'Build Executor Status' table shows one executor named '1' in a 'Idle' state.

#	W	Name	Last Success
1	☀️	2ASTG-SW	3 days 2 hr (28)
2	☀️	2ASTG-SW-BA	4 days 21 hr (23)
3	☀️	DEV-SW	3 days 9 hr (21)
4	☁️	DEV-SW-EHS-CB	5 days 3 hr (22)
5	☀️	DEV-SW-BA	4 days 20 hr (23)
6	☀️	DEV-EHS	5 days 2 hr (23)
7	☀️	DEV-EHS-BA	4 days 20 hr (22)
8	☁️	PROD-BC	4 days 22 hr (25)
9	☀️	PROD-BC-BA	4 days 20 hr (25)
10	☀️	PROD-SW	4 days 22 hr (21)
11	🌑	PROD-SW-EHS-CB	N/A

This screenshot displays the 'Process definition' page for 'ProcessRequest1.process'. It includes a description, a full process path, and a flow diagram. The diagram shows a sequence of steps: 'AppraisalRequestSub' (start), 'ReadXMLFile', 'ParseXMLFile', 'ReplyToAppraisal...', 'WaitForTaskConf...', 'SendAppraisalValue', 'Timeout', and 'End'. A 'WaitForConfirmation' step is also indicated below the diagram.

Section contains description of Process " ProcessRequest1.process " .  
Modification time: 2012/03/01 14:44:26

**Process description:**  
receive request for appraisal and get number of tasks from xml file

**Process definition:**  
Full process path: ProcessDefinitions/Appraiser1/ProcessRequest1.process

**Diagram:**

```
graph LR
    Start((AppraisalRequestSub)) --> ReadXMLFile[ReadXMLFile]
    ReadXMLFile --> ParseXMLFile[ParseXMLFile]
    ParseXMLFile --> ReplyToAppraisal[ReplyToAppraisal...]
    ReplyToAppraisal --> WaitForTaskConf[WaitForTaskConf...]
    WaitForTaskConf --> SendAppraisalValue[SendAppraisalValue]
    SendAppraisalValue --> Timeout[Timeout]
    Timeout --> End((End))
```

**Process starter activity:**  
Name: AppraisalRequestSub

- Property Name: **AppraisalRequestSub**
  - Type: *com.tibco.plugin.tibrv.RVEventSource*
  - Resource Type: *ae.rvpalette.RVEventSource*
  - Description: *subscriber for appraisal requests*
  - Configuration:
    - wantsValidationForOutput = true
    - wantsFiltrationForOutput = true
    - wantsXMLCompliantFieldNames = true
    - wantsRawOutput = false

# SUBSCRIPTION PLANS

## Developer



€4.000

- Eclipse GUI
- Individual License
- Desktop App

## Container



€5.000

- Headless mode
- Run in Docker or K8s
- Integrate into your pipelines

## Server



€12.000

- Web GUI
- Run in on-prem server
- Schedule and automate jobs
- Publish within intranet