

All the WealthSuite templates have the ability to deploy the following components.

Note: The version mentioned in the component filename (Eg: tap-db-golden-copy-201904.zip) is the T24 version and not the actual wealth component version. This pattern has been followed only to keep all the artifacts based on the T24 version within the artifactory.

Component description	Component name / type	Extension	Component structure and example	Comments	Mandatory component
Tap-core	tap-ta	zip	tap-ta-201904.zip ├── distribution_aaa_Linux.tar.gz ├── TA_Client_Documentation_Tools.zip	This is a mandatory Binary for the Wealth Template. Not applicable for the CLIENT.	Mandatory
	tap-ta-license	zip	tap-ta-license-201904.zip └── triplea_license_internal.imp	This binary defines the validity of the wealth supported products. Not applicable for the CLIENT.	Mandatory
OCS	tap-ocs	zip	tap-ocs-201904.zip ├── distribution_TAP.zip ├── documentation ├── install-client_TAP.tar.gz ├── legacy ├── pre_upgrade ├── release_info └── Update_Installer.jar	This is a mandatory component for the Front Office for the Wealth Template.	Mandatory
	tap-ocs-custo	zip	tap-ocs-custo-201904.zip ├── CUSTO-***-XXX-YYY.zip	This component is dependent on TAP OCS . There could be any number of custo files in the below pattern: CUSTO-***-XXX-YYY.zip → where XXX being the version number and YYY being the timestamp.	Optional
TTI	tti-config	zip	tti-config.-201904zip ├── brokerXML.properties ├── deploy.properties		Mandatory
	tti-deployer	zip	tti-deployer-201904.zip ├── ant_lib ├── common_tti.properties ├── deploy.xml ├── internal ├── macro.xml ├── queueNamesUpdateOnTTI-File.output.properties ├── queueNamesUpdateOnTTI.output.properties ├── sftp-file.output.properties ├── sftp.output.properties └── startAutoDeploy_laulxint3.sh		Mandatory
	tti-distribution	zip	tti-distribution-201904.zip ├── GWPACK ├── IB ├── OMS ├── T24 └── TAP		Mandatory
	tti-l3	zip	.. ├── tti-l3-201904.zip ├── BarFilesConfig ├── TTI.bar.properties ├── TTI-File.bar.properties ├── TTI-Log.bar.properties └── TTI-T24-Jms.bar.properties ├── BusConfig ├── aa-routing.xml ├── stmt-routing.xml ├── tti-rnfn.xml └── tti-flowmap.xml ├── Event_Flows.jar ├── XSLT ├── CustomerType ├── CustomerType-Custo.xml ├── CustomerType-Filter-Custo.xml ├── DataCaptureTransactionCashCredit ├── DataCaptureTransactionCashDebit-Custo.xml ├── DataCaptureTransactionCashDebit.xml └── L3Newflow ├── L3Newflow-Core.xml ├── L3Newflow-Filter-Core.xml ├── L3Newflow-Filter.xml └── L3Newflow.xml	Note that the L3 GWPACK format must follow the same structure as provided by Temenos! There's no L3 extraction mechanism provided by Product at this stage. While waiting for this product feature to be delivered, the client must maintain the L3 development in his repository mechanism, and publish packages to be testing in Temenos Continuous Deployment Platform. The Distribution must come as a single full distribution (i.e. any update/hotfix must be delivered in the Temenos Continuous Deployment Platform as a full release to be requested to Distribution team). BarFilesConfig Folder The bar file that needs to be deployed in l1B should have corresponding bar properties file naming convention. Eg. for the TTI-T24-Jms.bar.properties file: temenos.tti.connectors.B2F_T24toIN#JMS Input.sourceQueueName=jms/queue/t24IFInboundQueue These properties would be overwritten in the Bar file and redeployed. BusConfig Folder This folder contains properties and configuration files of TTI deployment. Event&Flows.jar folder This folder contains L3 updates for T24, specifically T24 DS. The L3 update is of jar type which will be compiled and published in T24 upon deployment. If there are no T24 updates, this folder can be left empty XSLT -- CustomerType CustomerType folder can consists of xsl files which are present both in L3 package and in L1 deployment. This should be of format "FileName-Custo.xml" XSLT -- L3Newflow L3Newflow folder can consists of xsl files which are present only in L3 package and not in L1 deployment.	Optional
GWPack	gwpack-config	zip	gwpack-config-201904.zip ├── env.properties ├── GWPACK_autoinstall.properties		Mandatory
	gwpack-deployer	zip	gwpack-deployer-201904.zip ├── ant_lib ├── autoActiveF2B.sh ├── autoActiveMQ.sh ├── autoGWPACKinstall.sh ├── common.properties └── deployer_gwpack.xml		Mandatory
	gwpack-distribution	zip	gwpack-distribution-201904.zip ├── gwpack_2.1.4_20200216_8.4.1.3.tar.gz └── GWPACK_Install.sh		Mandatory
	gwpack-l3	zip	└── gwpack-l3-201904.zip ├── GWP ├── Files ├── GWPACK.cfg └── mrm.csv └── Scripts └── 01_GWPACK_TTI_Att.sh	L3 GWPACK format must suit exact format to be provided by Temenos There's no L3 extraction mechanism provided by Product at this stage. Waiting for this product feature to be delivered, client must maintain L3 development in his repository mechanism, and publish packages to be testing in TCD. The Distribution must come as a single full distribution (i.e. any update/hotfix must be delivered in the Temenos Continuous Deployment Platform as a full release to be requested to Distribution team). mrm.csv - In this file, WTX executable to be used has to be updated with variable and value parameters ending with semicolon. It is updated for each line with semi colon separator in the CSV file (variable:variable;) . Eg: GWPACK_ONLINE_IMPORT_OPTIM:-m -O -B100; GWPACK.cfg - Customization required in the environment as part of L3 is listed in GWPACK.cfg file 01_GWPACK_TTI_Att.sh - This contains the Shell scripts which will be executed on top of GWP L1 deployment. Shell scripts execution sequence will be based on shell script name 01_{*}.sh, 02_{*}.sh. Eg: 01_GWPACK_TTI_Att.sh, 02_GWPACK_TTI_Att.sh etc.	Optional

Tap-Wsc	tap-wsc		tap-wsc-201904.zip ├── Channels_Installer_TechnicalGuide.pdf ├── Channels_IRIS_InstallationGuide.pdf ├── Channels_PM_InstallationGuide.pdf ├── current ├── FOPM.war ├── TCBIInstaller.jar └── WSC-FOPM-ODATA-25.10.0.zip	Not applicable for the CLIENT	Mandatory
ICT Cloning Tool	tap-cloning-tool	zip	tap-cloning-tool-201904.zip ├── antsrc ├── bin ├── data_extracted ├── env-clone.bat ├── env-clone.properties ├── env-clone.sh ├── lib ├── libapp ├── log4j.properties ├── logs ├── shell ├── sql └── temp	This is a IN house cloning tool used for the CLIENT Wealth Sybase based Templates for reducing the deployment tin Not applicable for WealthSuite monthly release templates, but mandatory for the CLIENT.	Optional
TAP DB Golden Copy	tap-db-golden-copy	zip	tap-db-qolden-copy-201904.zip ├── aaalogindb.dmp ├── aaamaindb.X.dmp ├── aaarepdb.dmp ├── aaatmptsldb.dmp ├── aaatsldb.X.dmp ├── env-clone_AAA_db.tar ├── goldencpy_env-clone_AAA_db.tar ├── master.dmp ├── model.dmp ├── sybsystemdb.dmp └── sybsystemprocs.dmp	This is the Golden Copy Provided By the Client for the ICT Tool to use it for the deployment Not applicable for WealthSuite monthly release templates, but mandatory for the CLIENT. Naming convention for specific files: aaamaindb.X.dmp aaatsldb.X.dmp → 'X' being the number of stripes, it can be from 1 to n number of stripes	Optional
TAP App Golden Copy	tap-app-golden-copy	zip	tap-app-qolden-copy-201904.zip ├── env-clone_AAA_app.tar.gz	This is the Golden Copy Provided By the Client for the ICT Tool to use it for the deployment Not applicable for WealthSuite monthly release templates, but mandatory for the CLIENT	Optional

Component description	Can this be client specific	How to change in an Extend environment	How to change in an Assemble factory
Tap-core	Yes	Client golden copy / ISB build must be uploaded into artifactory and the respective template manifest file (i.e. components file) updated with artifactory path	The client binary should be uploaded to the appropriate stream/stage folder in the artifactory. The exact location can be found in the TCD portal components configuration page. After creating components in the TCD portal, the upload URL will be populated automatically.
OCS	Yes	Same as above	Same as above
OCS-Custo	Yes	Same as above	Same as above
TTI	Yes	Same as above	Same as above
TTI -L3	Yes	Same as above	Same as above
GWPack	Yes	Same as above	Same as above
GWPack-L3	Yes	Same as above	Same as above
Tap-Wsc	Yes	Same as above	Same as above