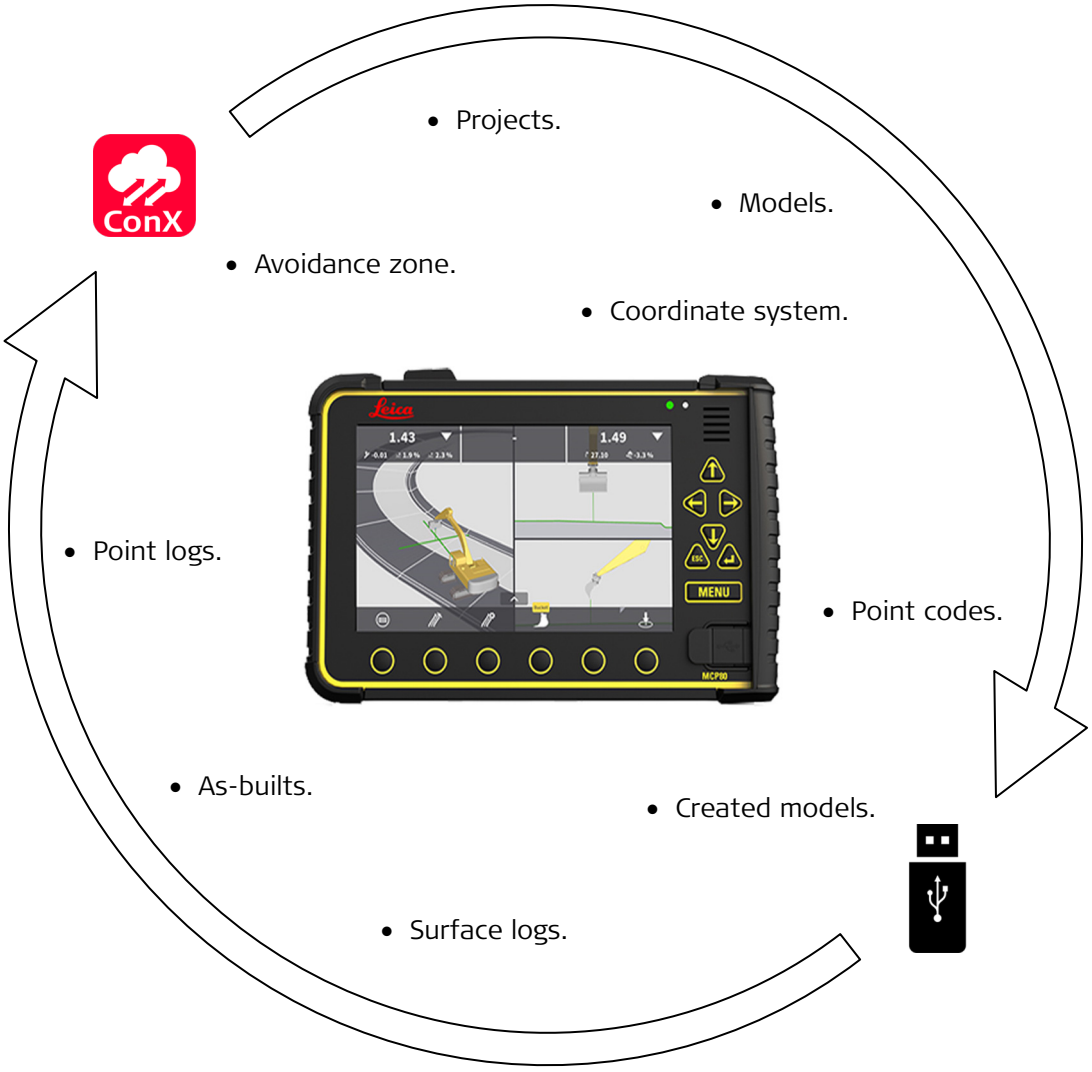
Data flow

User Manual

Leica MC1

Version: 1.0



PART OF HEXAGON

- when it has to be **right**

Leica Geosystems

Machine type: All machine types.

Manual # 892571

Software: MC1.

Manual type: User manual.

Manual version: 1.0

Made by: JEHA

Purpose: To describe data flow in relation to MC1.

Support: Leica Machine Control (LMC) support:

- Mail: lmc.support@leica-geosystems.com
- Phone: +45 70 224 226

Index

Introduction

The Leica machine control ecosystem consumes and produces large quantities of construction data. Some data is prepared and distributed online via ConX, some data is distributed via USB sticks and some data is created locally, on-site, via MC1 panel software.

Data relevant to the scope of this manual can be divided into four overall groups:

- **Design** (Active models, passive models, avoidance zones).
- **Localization** (Coordinate systems, geoid).
- **As-built** (Point logs, point codes, surface logs).
- **Maintenance** (System logs, crash dumps, updates, licenses, backups).

The data is organized and stored in folder structures - sometimes on more than one platform at the same time.

There are both software and hardware platforms, and it is the flow of data between these platforms that makes up the Leica machine control solutions.

This manual describes the four groups of data, as well as ConX (), USB () and MC1 (), all in a data flow context.

The manual aims to create an overview of how data is organized, and thus by extension how data interact.

Project specific data flow

A project can be defined mainly as a collection of **design** and **as-built** data related to a construction site.

If needed, **localization** data in the form of coordinates systems can also be included as part of a project.

Projects are the primary elements in MC1. Any machine operation using MC1 starts with a project, synchronized from ConX or a USB stick, or created directly in MC1 itself.

Projects can be accessed through MC1, ConX, USB and iCON Site. With the exception of a different update procedure, projects behave in a similar way no matter which platform has been used as means of access.

All project data can be found in folders (Project root folders) named after the project itself.

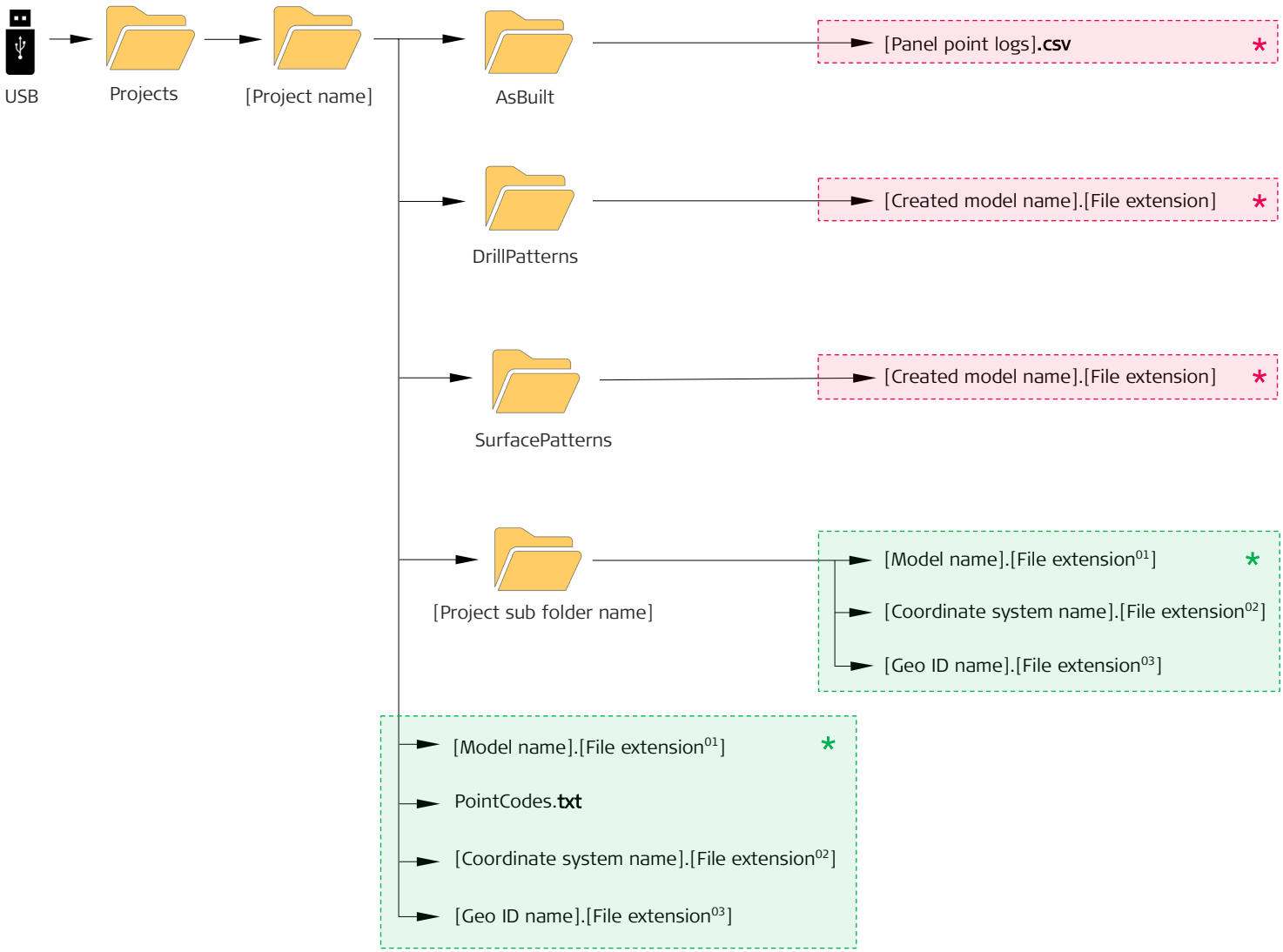
On a USB stick, project root folders are located within a parent folder called "Projects". On ConX, project root folders are accessed through a project tab in the user interface. Depending on project complexity, project root folders on both USB and ConX may contain additional project sub folders, structured into several levels.

Names and number of project sub folders are defined in the data preparation phase, and should not be a concern for the machine operator.

Projects connected to ConX are updated real time, enabling a continuous flow of new design and as-built data.

It is recommended **NOT** to mix project data from ConX with project data from USB (...or vice versa).

Project specific data flow (Continued)

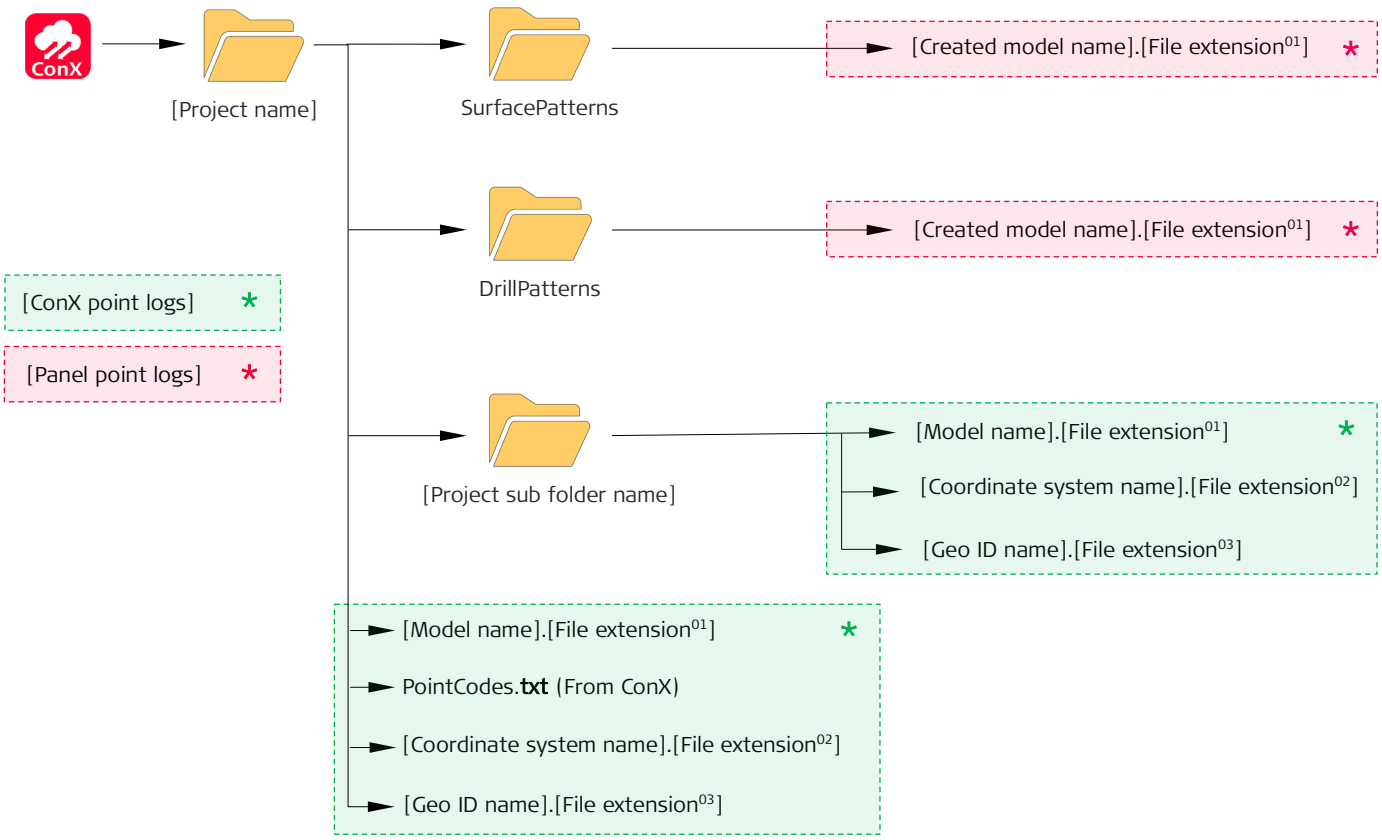


01 Model file extensions: **.dxf**, **.xml**, **.dxf**, **.geo**, **.ird**, **.kof**, **.l3d**, **.lin**, **.lmd**, **.mbs**, **.trm**

02 Coordinate system file extensions: **.xml**, **.lok**, **.dat**, **.dc** (Trimble), **.loc** (Carlson)

03 Geo ID file extension: **.grd**

Project specific data flow (Continued)



01 Model file extensions: **.dxf**, **.xml**, **.dxf**, **.geo**, **.ird**, **.kof**, **.l3d**, **.lin**, **.lmd**, **.mbs**, **.trm**

02 Coordinate system file extensions: **.xml**, **.lok**, **.dat**, **.dc** (Trimble), **.loc** (Carlson)

03 Geo ID file extension: **.grd**

Project specific data flow (Continued)

Design

In a project specific context, the "Design" section describes active and passive models, how they appear and behave on different platforms, and how they are used as avoidance zones.

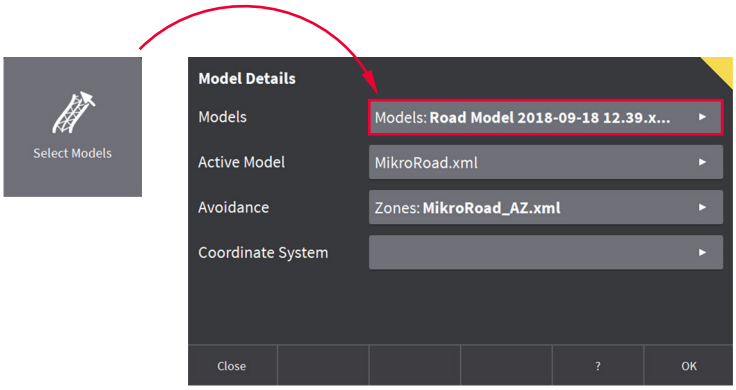
This section also describes file revisions before and after synchronization, exceptions regarding file formats and how tags are used to enable smart filtering of models in MC1 and ConX.

Passive models

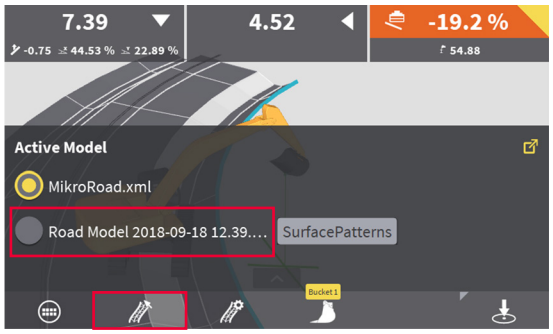
A passive model is in effect defined as any model that is not active – it is used as "background" with all objects (Points, lines and surfaces) visible, but not selectable.

A machine operator will not be able to work on any object in a passive model.

A passive model can only be viewed in the runscreen if it is part of the models selected in MC1 "Select Models" menu.



Selected models ("Select Models" menu)



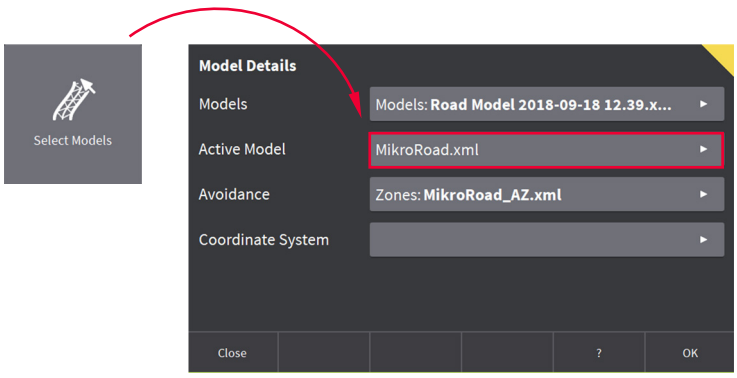
Passive model (Runscreen)

Active models

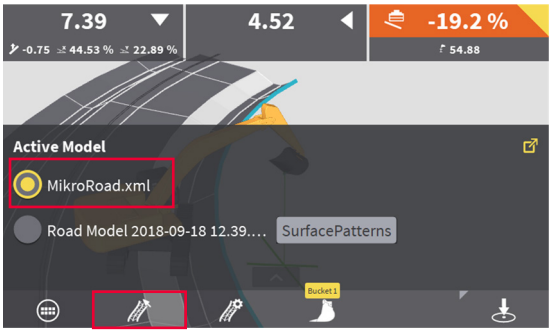
With the exception of distance to avoidance zones, all calculation in MC1, including creation of as-built data, is only done in relation to the currently active model.

Only one model at a time can be active in MC1 – all other models are defined as passive. Thus, an active model becomes passive the moment another model is defined as active.

The operator can define which model should be active, directly from the runscreen or in "Select Models" menu.



Active model ("Select Models" menu)



Active model (Runscreen)

Project specific data flow (Continued)

Avoidance zones

All supported model types, including 3D models, can be used when defining avoidance zones.

Avoidance zones can be based on passive models containing points, lines and surfaces.

The entire model is used as avoidance zone, and should initially be prepared as such.

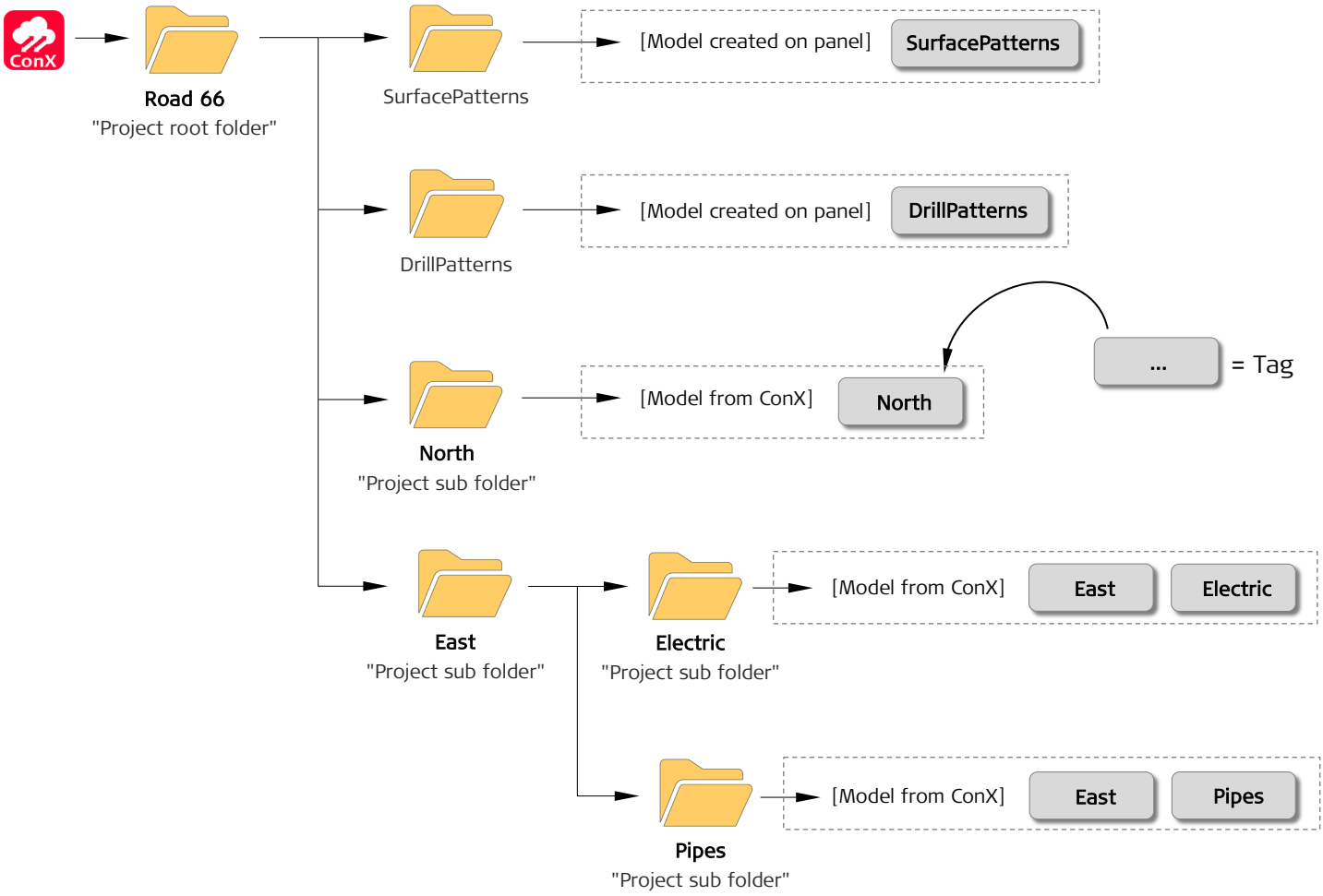
Model tags

Tags enable smart filtering of models in ConX and MC1.

Models are tagged when imported or synchronized with ConX. Un-synchronized models created in MC1, and models in USB projects, are **NOT** tagged.

Tags are named after the project sub folders in ConX, but to simplify data access for the machine operator, no folders are displayed on the panel. Instead, the machine operator can filter and thus find relevant models by selecting tags in MC1.

The number and hiearchy of tags attached to a model depends on the sub folders leading to where the model is stored (See illustration below). This means, that tags can be used as a bread crumb reflection of where a model can be found, and in what context it should be seen.



Project specific data flow (Continued)

Model appearance

Objects have the same appearance in all models, even if file types are different, making it easier for the machine operator to work with different models.

Plan view provides guidance based on current selections. It is recommended to use plan view in 2D mode when working with .dxf models. The reason for using 2D mode is that model files, including .dxf files, often contain objects with zero height.

Model behaviour

All models can be synchronized, including those created on the panel in MC1. Models created in MC1 are generated in open formats, making them easy to use outside the Leica ecosystem.

MC1 allows simultaneous use of a model reference for height calculation (Point, line and surface) and a model reference for side distance calculation (Line).

Choosing a single element, not an entire model, as height reference, always creates a surface through extention of the chosen element.

MC1 supports "corridors".

Use of corridors is a Leica concept, and indicates that two lines are connected by a virtual 3D surface.

Corridors behave just like triangulated surfaces. A triangulated surface can be used as height reference, but it is recommended to use corridors instead, as they will enable much faster and more accurate calculation.

Corridors effectively allow perfect grading of an area between two lines, without the need for triangulation.

Available corridor model formats: **LandXML**, **LMD** and **MBS**.

Exceptions

DXF files contain texts in labels that are not available in other files. This text information can be viewed in the 2D plan view.

DXF files contain layers of objects.

The machine operator can access all elements, but cannot select/deselect layers or turn layers on/off.

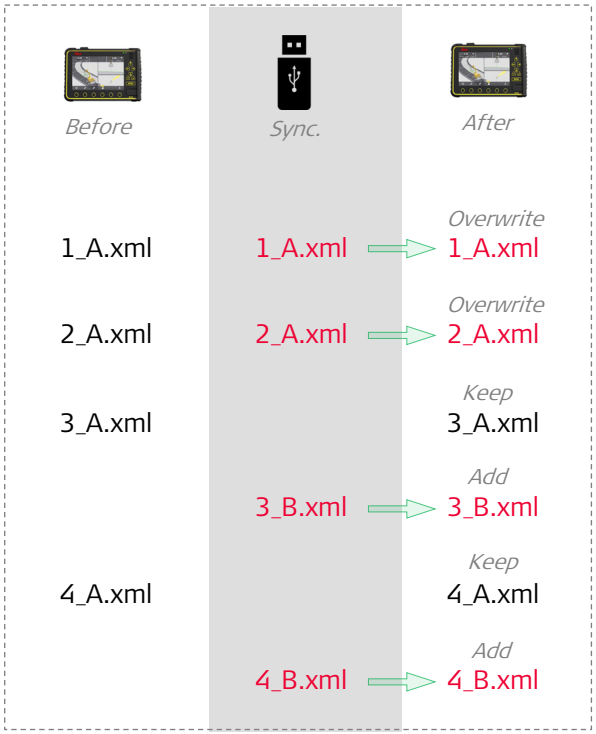
Project specific data flow (Continued)

Revisions

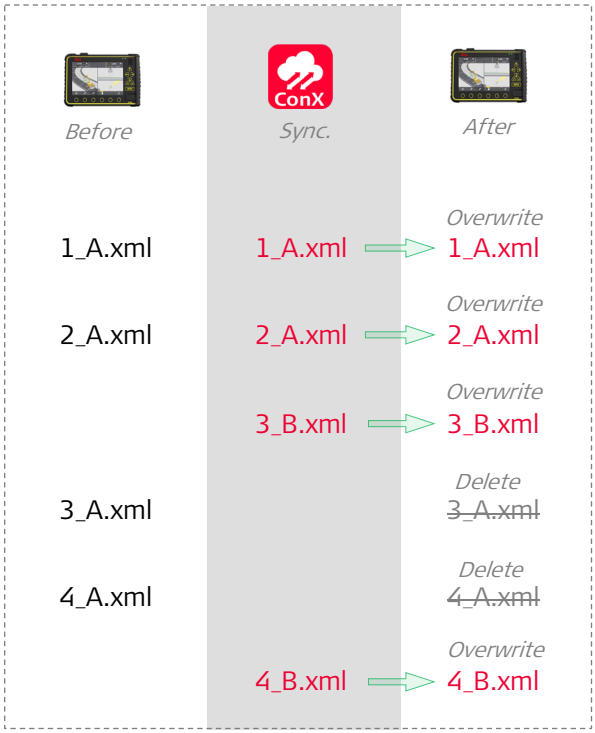
ConX is master, and after synchronization, files in MC1 will mirror files on ConX. A ConX synchronization effectively deletes files in MC1 that are not in ConX, and overwrites files with identical names. It is recommented to let ConX handle file revisions, and not keep multiple versions of the same file in ConX.

A USB synchronization overwrites files in MC1 with files of identical names from the USB stick. A USB synchronization does **NOT** delete files in MC1.

Files on the USB stick, or in ConX, will **NOT** be deleted or overwritten as a result of synchronization.



*File revisions before and after **USB sync.***



*File revisions before and after **ConX sync.***

Project specific data flow (Continued)

Localization

Coordinate systems

Coordinate systems can be stored **separate from** project data, **or** they can be stored **together with** project data to reduce operator involvement.

If stored separate from project data, on a USB stick, the coordinate systems are placed in predefined country specific folders. Initiating a USB synchronization in MC1 will update the panel with additional predefined coordinate systems from the country specific folders on the USB stick. If needed, to create a better overview, predefined coordinate systems on the panel can be deleted in MC1 "Maintenace" menu.

If stored together with project data, the coordinate systems are placed in the project root folder (Folder with project name), or in one of the project sub folders. Coordinate systems can be included in project root/sub folders in both USB and ConX projects.

A coordinate system in a project root folder covers all models in that project, with the exception of models sharing a sub folder with another coordinate system.

A coordinate system in a project sub folder only cover models in that specific sub folder.

In ConX projects, coordinate system and geo ID files are selected during project configuration. The coordinate system file[01] must be manually assigned to all machines. The geo ID file[02] will be assigned automatically along with the coordinate system file.

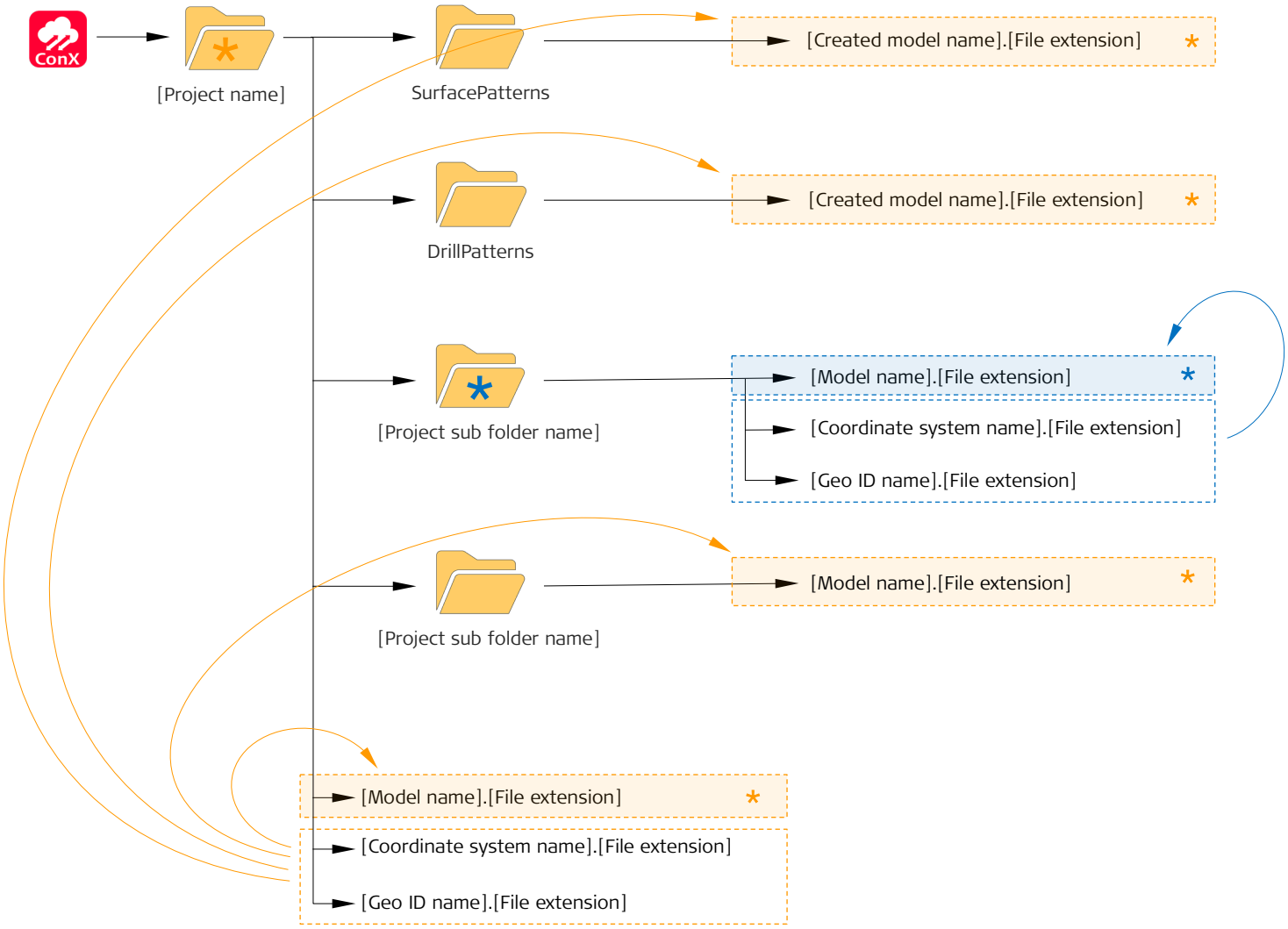
In USB projects, coordinate system and geo ID files are selected on-site in MC1. Neither the coordinate system file or geo ID file is assigned automatically, instead they must be selected manually from the predefined country specific folders. If the coordinate system carries geo ID information, then a geo ID file is suggested by MC1.

[01] Coordinate system file extensions: **.xml**, **.lok**, **.dat**, **.dc** (Trimble), **.loc** (Carlson)

[02] Geo ID file extension: **.grd**

Project specific data flow (Continued)

Coordinate systems in a ConX project:

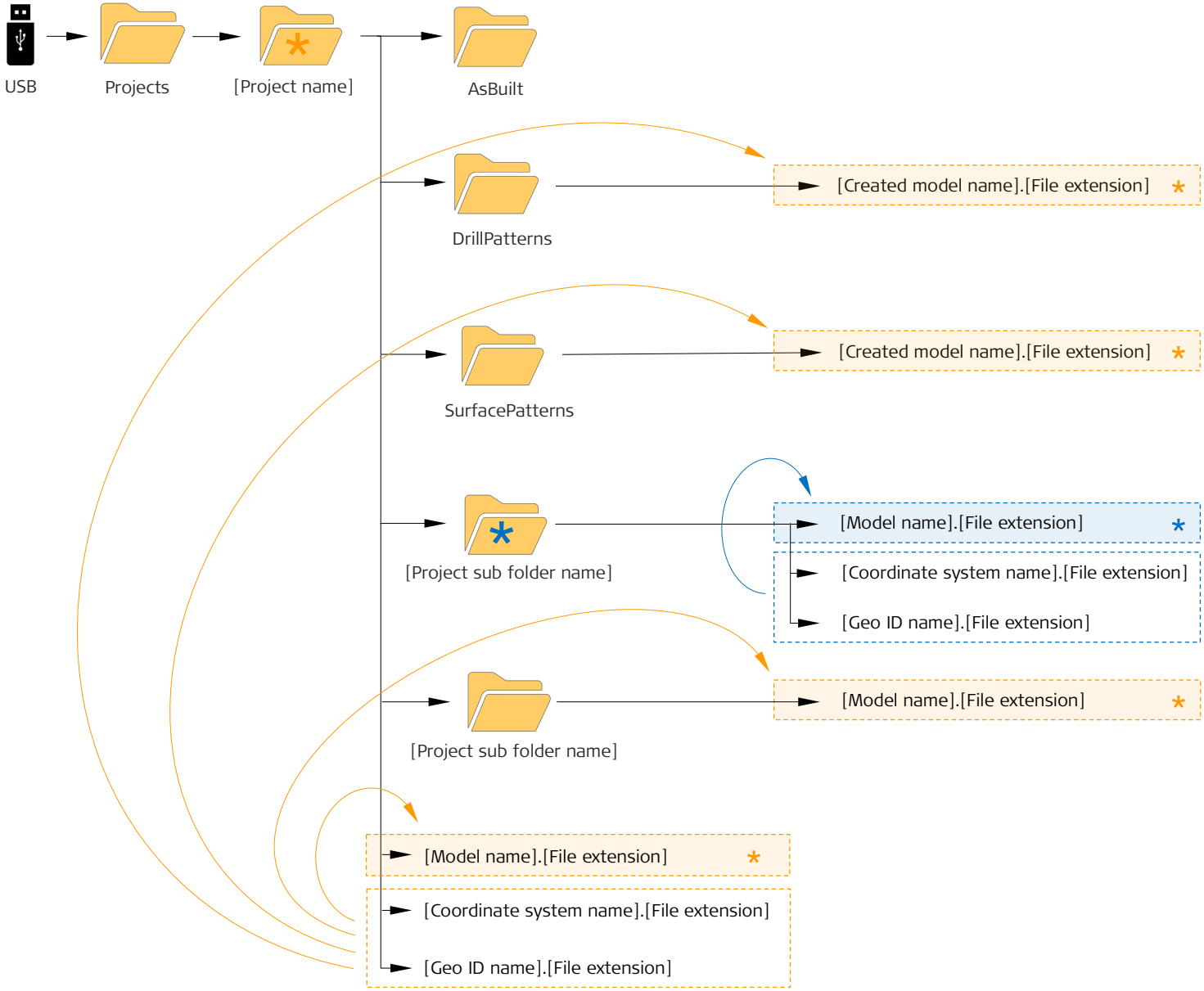


*Model/s covered by coordinate system in project sub folder.

*Model/s covered by coordinate system in project root folder.

Project specific data flow (Continued)

Coordinate systems in a USB project:



*Model/s covered by coordinate system in project sub folder.

*Model/s covered by coordinate system in project root folder.

Project specific data flow (Continued)

As-built

Project specific as-built data consists of point logs (Single point logging), point codes and surface logs (Multi point logging).

Point codes are integral parts of point logs:

- Point code = code + log name.
- Point log = ID + *code + log name* + position + date/time + auxiliary information.

All as-built data can be shared between machines using ConX.

Use cases at the end of this document provide an overview of how point logs and point codes are synchronized between USB and MC1, and between ConX and MC1.

Point logs

Points logged from the runscreen in a USB project are stored in a CSV file named after the active model:

- [Active model name].[Model file extension].csv

  Example: **HighwayE20.xml.csv**

When synchronized with a USB stick, the panel serial number is added to the CSV file name:

- [Active model name].[Model file extension].[Panel serial number].csv

  Example: **HighwayE20.xml.1234567.csv**

After USB synchronization the CSV file is located in the AsBuilt-folder on the USB stick:

- USB: \ Projects \ [Project name] \ AsBuilt \ [Active model name].[Model file extension].[Panel serial number].csv

  Example: **USB: \ Projects \ Leica Highways \ AsBuilt \ HighwayE20.xml.1234567.csv**

In the runscreen logged points are displayed as colored squares:

- My machine: (Purple)
- Other machines: (Grey)

Points logged from the runscreen in a ConX project are stored as raw data available for streaming between MC1 and ConX, as part of a ConX synchronization.

Project specific data flow (Continued)

Point logs (As .csv files or raw ConX data) can contain a large amount of information.

Below is a list of available information that will be associated with each logged point:

PointID	PointID indetifies a point log.
PointCode LogName	Point code = code (point code) + log name.
Easting Northing Height WGS84Longitude WGS84Latitude EllipsoidHeight GNSSCQ3D	Position data. (Accuracy of GNSS signal)
Date Time	Date and time of point log.
ConXMachineName PanelSerialNumber	Identifcation.
ActiveModel	Base for point log.
HeightReference HeightReferenceType HeightOffsetType HeightOffset HeightDistance	Vertical data related to model references.
SideReference SideReferenceType SideOffset SideDistance	Horizontal data related to model references.
Station StationingReference CenterLineDistance	Stationing data.
ToolName	Tool definition.

Project specific data flow (Continued)

Point codes (Project)

Point codes identify points of interest. In combination with a log name, point codes provide a simple identification, and definition, of the primary characteristic related to a specific location in a model. Whenever a point is logged from the MC1 runscreen, a preselected point code is associated with that specific point.

The point codes available for use in MC1 are listed in a text file named "PointCodes.txt". There are two kinds of code lists, both of which are named "PointCodes.txt".

One of the code lists is a "project" specific code list, and the other is a "machine" specific code list.

During data preparation, relevant point codes, originatinating from various machines, projects, authorities, contractors etc. are compiled into a project code list. This list is made available for MC1, on the panel, through either ConX or a USB stick. Typically the project code list is compiled by the site manager.

In ConX and on a USB stick the project code list is stored in the project root folder, together with models and coordinate systems.

The machine operator cannot add, delete or in any way modify codes on the project code list.

There is only one project code list per project.

Surface logs

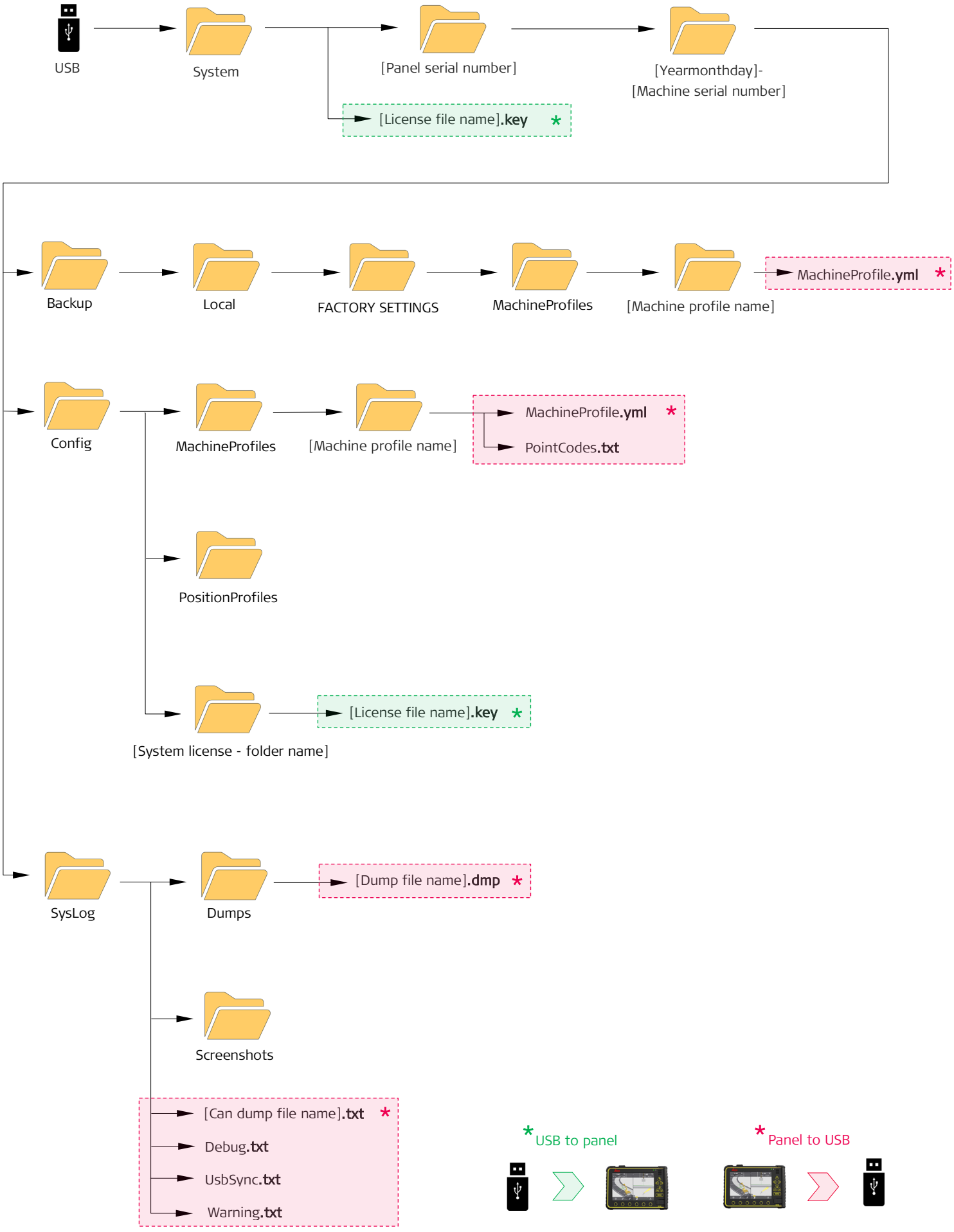
Surface logs are basically grids of data, based on the currently active surface.

Depending on the type of machine, surface logs are used to document different surface characteristics:

- Snowgroomer: Snow depths.
- Compactor: Compaction degree and pass counts.

Surface logs can be shared between machines, and reports can be extracted, through ConX.

Machine specific data flow



Machine specific data flow (Continued)

As-built

Point codes (Machine)

Point codes identify points of interest. In combination with a log name, point codes provide a simple identification, and definition, of the primary characteristic related to a specific location in a model. Whenever a point is logged from the MC1 runscreen, a preselected point code is associated with that specific point.

The point codes available for use in MC1 are listed in a text file named "PointCodes.txt". There are two kinds of code lists, both of which are named "PointCodes.txt".

One of the code lists is a "project" code list, and the other is a "machine" code list.

If point codes contained in the project code list are found to be insufficient, then new point codes can be created. New point codes are created directly on the panel, typically by the machine operator.

On a USB stick, the machine code list is stored in a machine profile folder located within the "Config" folder, which furthermore is located within the "System" folder (See illustration in previous section).

Prior to synchronization, point codes on the machine code list are only available on the panel where it was created, and only in the machine using that specific panel.

The machine code list can be shared between machines using ConX.

There is only one machine code list per machine.

Machine specific data flow (Continued)

Maintenance

Maintenance is defined by system logs, crash dumps, updates, licenses and backups.

System logs

A system log is a log of all errors, warnings and information messages at any given time and date.

Crash dumps

Crash dumps are used, in rare cases, to find errors on a system.

Crash dumps are created and sent to ConX automatically, but can also be synced to USB.

Updates

Here both software for GNSS receiver and Panels can be managed.

For fast upgrade of iCG80, insert a USB stick into iCG80 and initiate upgrade from MC1.

Licenses

Here both licenses for GNSS receiver and Panels can be managed.

Licenses are stored as a key files (**.key**).

The location of key files in USB or ConX folder structures can vary from project to project.

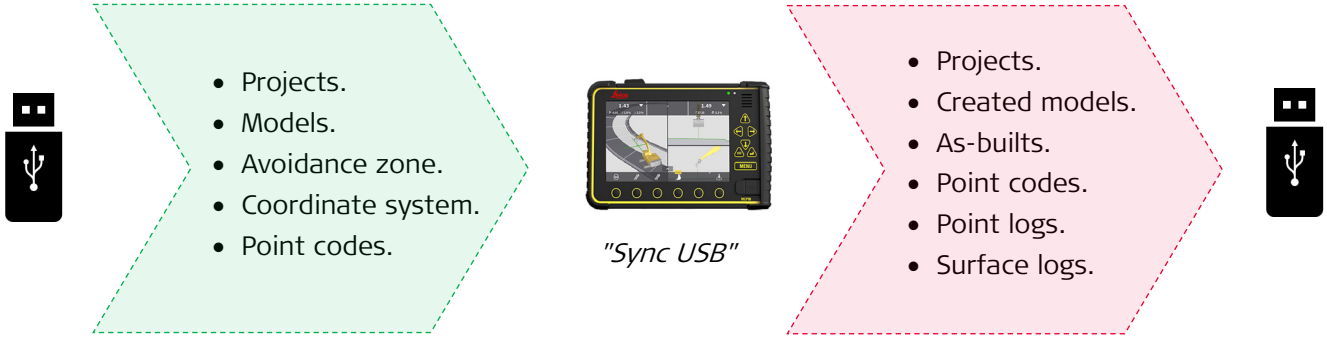
Backups

Backups are synchronized automatically to a USB stick, if such is inserted into the panel.

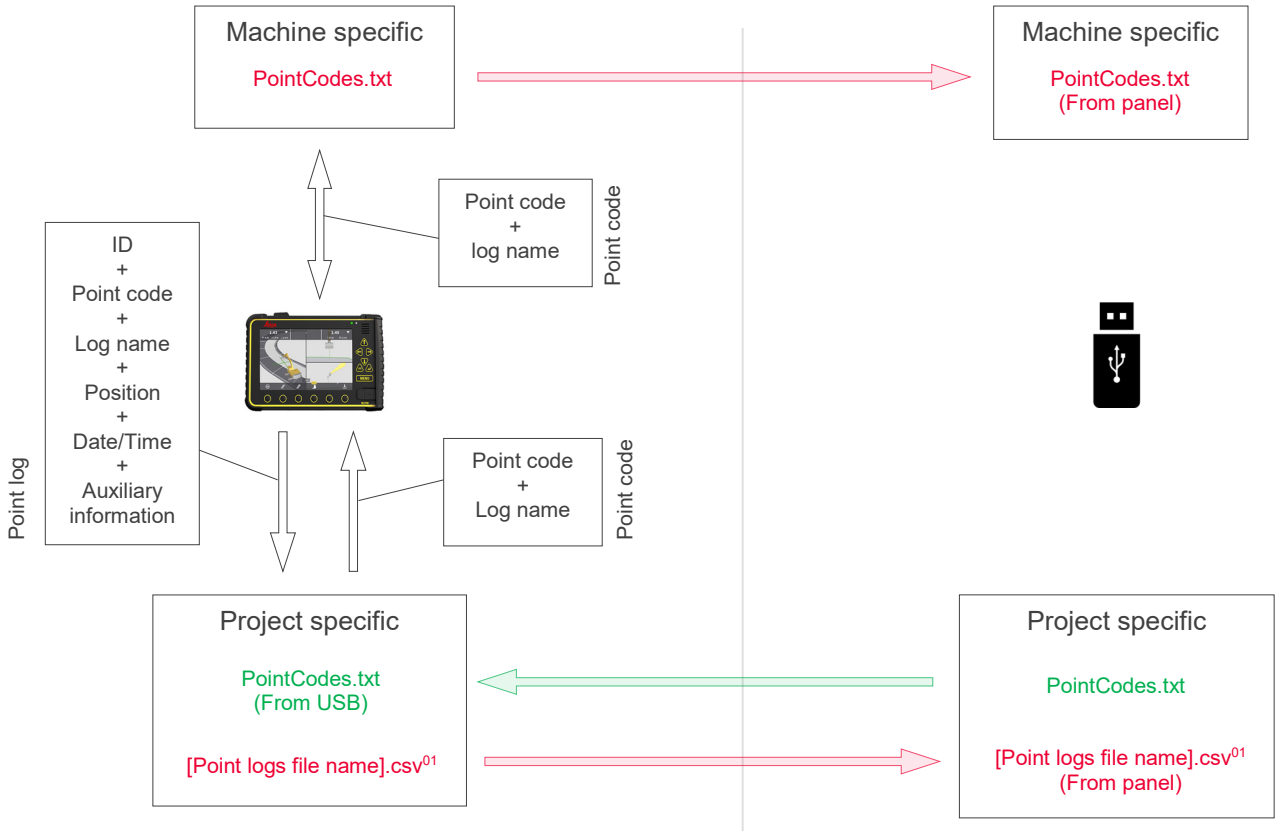
It is possible to restore backed up data from USB or ConX.

Use cases

USB – MC1 – USB

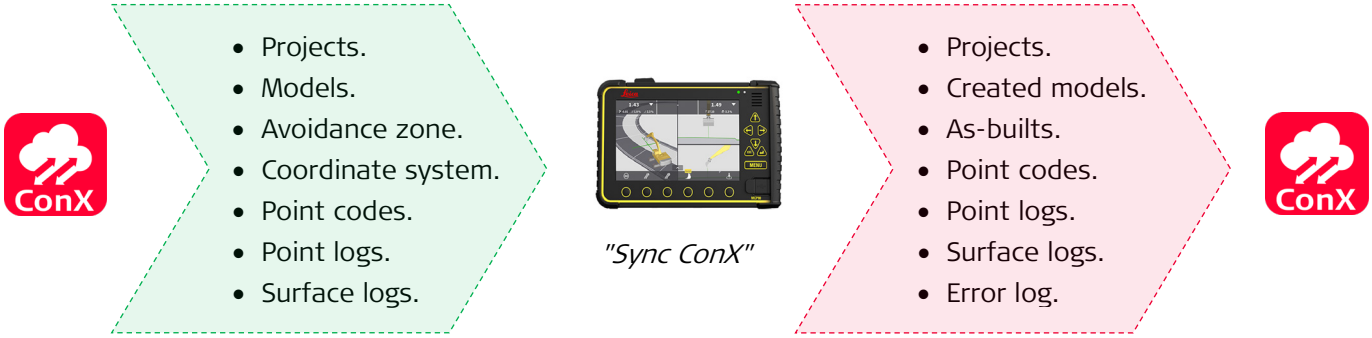


Point code / point log synchronization



01 List of panel point logs: [Active model name].[Model file extension].[Panel serial number].csv

ConX – MC1 – ConX



Point code / point log synchronization