# Utilisation Analysis

## Understanding Asset Usage

# Overview

## Abstract

This document introduces ARDIs Utilisation addon and covers some of the cases where it might be used.

## Assumed Knowledge

None.

## Who to Talk To?

For more information please contact Optrix.

## Contents

# Introduction

## What is Utilisation?

**Utilisation** measures how particular types of asset (such as machinery, robots, vehicles etc.) are being used over time.

At it's most simple, utilisation tells you how many hours a day a particular machine has been running, moving, or otherwise being used.

But you can **much** more insight out of utilisation data by combining your data sources to create a more detailed view of asset status.

## Why Would I Need It?

Understanding how your assets are being utilised helps in a number of ways.

The strongest benefit is how it helps to **find opportunities to improve a process**.

It does this by highlighting the times – and in some cases *causes* – of availability issues in your system, such as bottlenecks, holdups, downtimes and failures.

It's also useful for ensuring workloads are being evenly distributed across your assets, which is important if you are trying to optimise your maintenance and reduce wear, tear and exhaustion on your assets.

# How It Works

## Defining Accurate Asset States

First, you define what *states* each of your assets can be in.

This can combine information from multiple different information management systems, such as using downtime information with GPS data and machine sensors.

This allows you to create an 'asset state' that is more accurate than you can create using sensor data alone. Instead of just being 'Stopped' and 'Running', you can indicate *why* and *where* issues might be occurring.

For example, rather than just a simple "Stopped" state, a piece of mobile equipment can be shown as 'Stopped for Collision Avoidance @ Building B Entrance' or 'Stopped for Scheduled Maintenance @ Maintenance Bay 2'.

These more detailed statuses allow you to understand much more about what assets are doing, where they were located at the time, and why they were in that state. By adding more detail, we get a much clearer idea of where problems lie and where improvements can be made.

## Defining Cycle Times

You can also optionally define *cycles* – this allows the system to measure the total time it takes assets to perform routine tasks.

For example, a production line creating a batch of product. Or a vehicle performing a full routine of loading product, moving, unloading and returning.

Analysing cycle times allows you to identify individual assets that are performing less efficiently and review why this might be occurring.

## Easy Interface

Once these are defined, the addon provides an easy, point-and-click interface that lets you explore your results visually, with dynamic, interactive web-based charts that allow you to deep-dive into your data.

## API Access

Details are also available via API functions, allowing you to use utilisation data in your own solutions – such as reports or OEE calculations.

This gives you a single, centralised source of information. It also means that if you need to change your logic in the future – for example, by adding new sensors and machines – you only need to update a single configuration rather than changing your utilisation logic in many different places.

# Use Cases

## Identify Bottlenecks

In our vehicle movement example, we have several vehicles that drive underground to pick up product and deliver it to a storage area on the surface.

However, our roads are quite narrow and many only fit a single vehicle at a time. As such, they can be congested – vehicles end up waiting for others.

If we wanted to improve our overall efficiency, where could we look?

Let's start with the overall picture…

The **green** areas represent normal operation. **Grey** areas are normal downtimes – such as shift changes and waiting times.

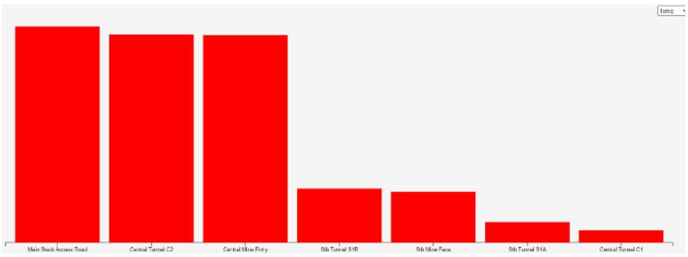
**Blue** and **purple** are loading and unloading operations.

**Red** times show where vehicles are queued and waiting for others to pass.

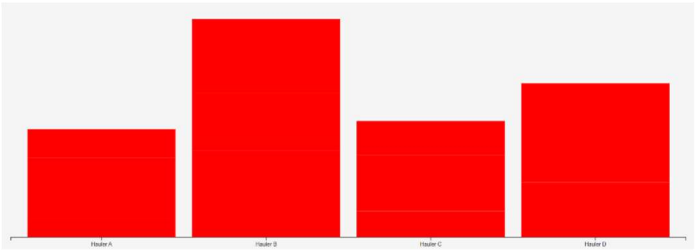
From the chart above, we can see that operations are quite good – the bulk of the time, things are operating well.

We are spending quite a bit of time in the grey area (downtime). Some is part of normal operation, such as shift changes. Some (marked as 'Returning' and 'Crib Room') is because there isn't anything for the drivers to do. On this site, that's acceptable – they like to have spare capacity.

But there is a small percentage of the time where our vehicles are queued up waiting. We can click on those times to zoom in.



Here we can see that most of the time spent waiting are in the *main stack access road* and *central mine entry*. This tells us that expanding the entrance or adding an extra lane to offload product could result in better production.



We can also find which of our assets are being impacted the most by these issues. Right now, *Hauler B* is the one that spends the most time waiting.

Utilisation

## Identify Loss in Asset Performance

To continue the example above, we'd like to review how efficiently each of the machines are being used *individually*.

To do this, we can look at their **cycle timing**. Increases in cycle timing over time indicate issues that may need to be addressed.

These vehicles are intended to head down into an underground mine, collect the resources (called *loading*), drive back to the surface and then drop off the load (*unloading*).

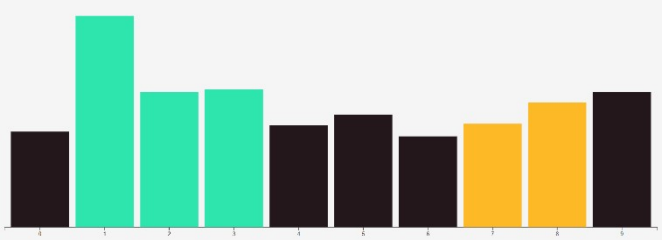
So we can define a 'cycle' as the full time it takes between unloading operations.

When performing this sort of analytic, it's always good to consider *fairness*. In this example, we're going to be focusing on a process that involves human drivers. If we're measuring their performance, we should consider if there's any times we *shouldn't count* as part of the cycle time.

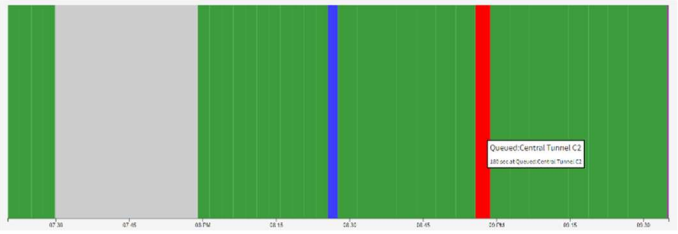
We obviously want to include any times where the vehicle is travelling, loading and unloading. We also want to consider times when the vehicle ends up queued behind another.

But should shift-changes and times when there isn't any product to be loaded be included in these metrics? Generally, we'd say 'No' – that isn't the fault of the driver or the equipment, so it's best that those times don't influence the metrics.

So we can set **delay** times to be ignored when calculating cycle times, ensuring that those times spent changing drivers or waiting for new jobs won't count against the metrics for an asset.



Once our rules are set up, we can easily see a summary of the recent cycle times. In this case, they're colour-coded based on the destination, as the vehicles pick up from one of three underground locations. As you can see, the teal-coloured destination (the 'North Loading Bay') takes more time on average than the other destinations.




A more detailed view of one of these cycles shows what the vehicle was doing.

It returned to the parking area, waited for a new job to come in, then went underground. Loading was uninterrupted, but the vehicle did get held up waiting in the Central Tunnel.

Utilisation