Developing sustainable processes for IATI import

Development partners have published a large amount of data on their aid activities through the International Aid Transparency Initiative (IATI). We’re going to start using IATI data in Bangladesh’s Aid Information Management System (AIMS).

The aim is to improve the quality of the data held in the AIMS and reduce the burden of data collection.



Technical Assistance By



TechnoVista Limited

Table of Contents

[Developing sustainable processes for IATI import 1](#_Toc453857984)

[Background 8](#_Toc453857985)

[AIMS overview 8](#_Toc453857986)

[Background 8](#_Toc453857987)

[Comparison between AIMS and IATI data 9](#_Toc453857988)

[Surrounding processes 10](#_Toc453857989)

[Technology used in the AIMS 11](#_Toc453857990)

[References 12](#_Toc453857991)

[Overview and core resources 12](#_Toc453857992)

[Pilots to date 12](#_Toc453857993)

[Useful code 12](#_Toc453857994)

[Data 13](#_Toc453857995)

[Analysis of specific donors’ data 14](#_Toc453857996)

[Canada 14](#_Toc453857997)

[Considering how to handle trust funds etc. 15](#_Toc453857998)

[Example project 15](#_Toc453857999)

[DFID 15](#_Toc453858000)

[Project components 15](#_Toc453858001)

[Implementing organisations 16](#_Toc453858002)

[Germany 16](#_Toc453858003)

[Project components 16](#_Toc453858004)

[World Bank 17](#_Toc453858005)

[Other donors 17](#_Toc453858006)

[Netherlands 17](#_Toc453858007)

[UNDP 17](#_Toc453858008)

[UNICEF 18](#_Toc453858009)

[USAID 18](#_Toc453858010)

[Development 19](#_Toc453858011)

[Development process 19](#_Toc453858012)

[Indicative development roadmap 19](#_Toc453858013)

[Sprint 1 20](#_Toc453858014)

[Overview 20](#_Toc453858015)

[Data integrity 20](#_Toc453858016)

[DP data overview 21](#_Toc453858017)

[1. Retrieve data from IATI Datastore 21](#_Toc453858018)

[2. Conversion 22](#_Toc453858019)

[3. Storage 22](#_Toc453858020)

[4. Make data available to rest of application 24](#_Toc453858021)

[5. Read data from the AIMS in the IATI Standard 24](#_Toc453858022)

[6. Write data to the AIMS 25](#_Toc453858023)

[Sprint 2 25](#_Toc453858024)

[Overview 25](#_Toc453858025)

[Data integrity 26](#_Toc453858026)

[DP data overview 26](#_Toc453858027)

[1. Build a basic interface 26](#_Toc453858028)

[2. Filter and group projects 27](#_Toc453858029)

[3. Match projects between IATI and AIMS 28](#_Toc453858030)

[4. Display matched and unmatched projects 28](#_Toc453858031)

[Sprint 3 29](#_Toc453858032)

[Overview 29](#_Toc453858033)

[Data integrity 29](#_Toc453858034)

[DP data overview 29](#_Toc453858035)

[1. Build a general field selection interface 30](#_Toc453858036)

[Storing preferences 30](#_Toc453858037)

[2. Display a list of projects 31](#_Toc453858038)

[3. Project-level field preference selection 31](#_Toc453858039)

[4. Test import of basic data fields to the test AIMS system 31](#_Toc453858040)

[5. Think about how to handle financial data 31](#_Toc453858041)

[Sprint 4 32](#_Toc453858042)

[Overview 32](#_Toc453858043)

[Data integrity 32](#_Toc453858044)

[DP data overview 33](#_Toc453858045)

[1. Allow users to pass activities over to each other 33](#_Toc453858046)

[2. Development of interfaces to handle more complex projects 34](#_Toc453858047)

[3. Trust Funds “overview interface” 34](#_Toc453858048)

[4. Co-financing “overview interface” 34](#_Toc453858049)

[5. User management 35](#_Toc453858050)

[Sprint 5 35](#_Toc453858051)

[Overview 36](#_Toc453858052)

[Data integrity 36](#_Toc453858053)

[DP data overview 36](#_Toc453858054)

[1. Import basic project data 36](#_Toc453858055)

[2. Import sectors 38](#_Toc453858056)

[3. Import implementing organisations 39](#_Toc453858057)

[4. Import project documents 40](#_Toc453858058)

[5. Import locations 41](#_Toc453858059)

[6. Handle updates 41](#_Toc453858060)

[Sprint 6 41](#_Toc453858061)

[Overview 42](#_Toc453858062)

[Data integrity 42](#_Toc453858063)

[DP data overview 42](#_Toc453858064)

[1. Collect and display IATI financial data 43](#_Toc453858065)

[2. Inserting financial data to AIMS - first time 44](#_Toc453858066)

[3. Handling updates of IATI financial data 45](#_Toc453858067)

[4. DP dashboard 46](#_Toc453858068)

[Update: first week of sprint 6 47](#_Toc453858069)

[Update: second week of sprint 6 47](#_Toc453858070)

[Sprint 7 48](#_Toc453858071)

[Overview 49](#_Toc453858072)

[Data integrity 49](#_Toc453858073)

[DP data overview 49](#_Toc453858074)

[1. Allow components of projects to be deselected 49](#_Toc453858075)

[2. Improve manual mapping interface 49](#_Toc453858076)

[3. Allow projects to be grouped 50](#_Toc453858077)

[4. Save progress as the user works through the stages 50](#_Toc453858078)

[5. Speed up performance 50](#_Toc453858079)

[Remaining to do 50](#_Toc453858080)

[Documentation 52](#_Toc453858081)

[Retrieving IATI data 52](#_Toc453858082)

[Introduction 52](#_Toc453858083)

[Retrieving from the IATI Datastore 52](#_Toc453858084)

[Troubleshooting 52](#_Toc453858085)

[Hierarchies of projects 52](#_Toc453858086)

[Understanding hierarchies in IATI 53](#_Toc453858087)

[Relating this to the AIMS 53](#_Toc453858088)

[How to apply this to the AIMS 54](#_Toc453858089)

[Importing this data 54](#_Toc453858090)

[Project types 54](#_Toc453858091)

[Financing relationships in the Bangladesh AIMS 54](#_Toc453858092)

[Distingushing between types of projects in IATI data 55](#_Toc453858093)

[Financial data 56](#_Toc453858094)

[Financial data in the AIMS 56](#_Toc453858095)

[Financial data in IATI 57](#_Toc453858096)

[Mapping financial data from IATI to AIMS 57](#_Toc453858097)

[Updating data 59](#_Toc453858098)

[Updating activities 59](#_Toc453858099)

[Updating transactions 60](#_Toc453858100)

[Versions in IATI data 60](#_Toc453858101)

[Versions of the IATI Standard 60](#_Toc453858102)

[Approach to handling different versions 61](#_Toc453858103)

[Standardising and converting data 62](#_Toc453858104)

[Modifications to the AIMS 62](#_Toc453858105)

[Reading from the AIMS 62](#_Toc453858106)

[Writing to the AIMS 63](#_Toc453858107)

[Merging and updating cofinanced projects 63](#_Toc453858108)

[Collecting and storing mappings 63](#_Toc453858109)

[Collecting and storing preferences 64](#_Toc453858110)

[Displaying preferences 64](#_Toc453858111)

[Currency conversion 65](#_Toc453858112)

[Currency conversion rates to BDT 65](#_Toc453858113)

[Currency availability in central bank data 66](#_Toc453858114)

[Other currencies 66](#_Toc453858115)

[Glossary 66](#_Toc453858116)

# Background

## AIMS overview

### Background

Launched in October 2014, the Bangladesh AIMS has been designed by local staff based on their needs and after reviewing several other AIMS systems. The design was implemented by Technovista, a local software development firm.

The AIMS collects project and financing agreement level data in a comparatively comprehensive system. The AIMS is able to record financial data for commitments, planned disbursements, actual disbursements and expenditure. The AIMS allows for reporting on trust funds and multi-donor projects, can handle multiple currencies for all transactions, sub-national location data, both thematic and sector priorities (according to the government’s national development plan), aid effectiveness indicators, and project documents. Some fields are mandatory, but most are optional. There are a variety of pre-made reports and charts, all also downloadable and options to export custom data selections. Many fields are adjustable via the interface – thematic priorities, geographic areas, currency conversion rates and document types can all be modified without needing a developer to reprogram the AIMS.

Several aspects stand out with respect to IATI integration:

* consideration has already been given to providing an “API to ensure interoperability with other as such systems of the Government”;
* the AIMS was designed with IATI data in mind, therefore wherever possible, taking IATI data definitions and structures as the starting point in an effort to make it simple to incorporate IATI format data;
* projects have an optional field for the IATI activity identifier, which could facilitate matching of projects;
* source code is owned by the government and can therefore be adjusted without having to request permission from software vendors;
* some of the mapping work between IATI codelists and local codelists has already been done.

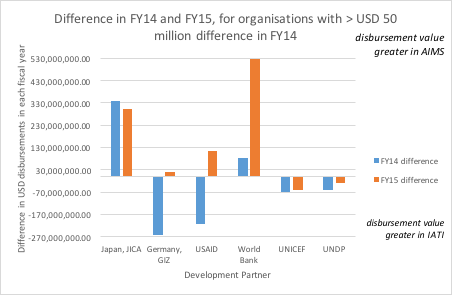
DPs are responsible for entering data in the system. Since the launch, 47 DPs have provided data into the system. Over time it is expected that more detailed project data is provided; there are a number of DPs that have yet to input any data to the system.

Data entry for a project involves filling some 40 fields even excluding project documents, results information or aid effectiveness indicators. Each financial transaction involves providing another 5-10 fields. Development partners report that this takes about 10 minutes per project. Some DPs have indicated that the data entry burden is partly responsible for the lower levels of data and lower data quality. These issues are likely to be compounded where DPs are non-resident, or where ERD has limited contact with them. As a result, ERD staff have also been carrying out significant data entry on behalf of DPs.

### Comparison between AIMS and IATI data

In October 2015 the AIMS underwent a substantial exercise in data collection in advance of the Bangladesh Development Partner Forum. A snapshot was taken on the 17th November 2015. AIMS data was compared to IATI data from http://dportal.org taken on the same day. Both AIMS and IATI contain significant amounts of data, AIMS reporting FY14 disbursements of USD 2.841 billion and IATI reporting USD 3.156 billion for the same period. The aggregate numbers are broadly similar, but the breakdown tells a somewhat more complex story. The AIMS records 46 organisations and IATI records 88 organisations. 11 of the organisations reported to the AIMS are not reported to IATI. AIMS data and IATI data therefore appear to be highly complementary. There are also significant variations between what is published to IATI and what is reported to the AIMS for specific DPs. Chart 1 shows all organisations for whom the difference in volume between the AIMS and IATI is greater than USD 50 million.

#### Chart 1: Comparison of actual disbursements recorded for FY14 and FY15

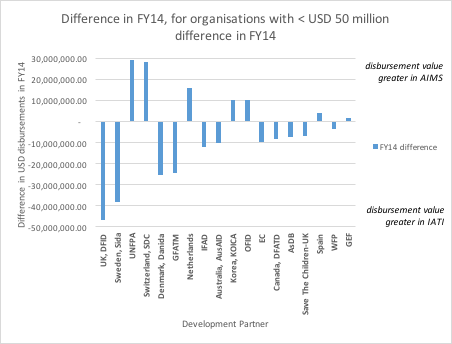


Where there is a difference between data in the AIMS, investigation is needed to determine the cause. The differences in GIZ are attributable to the issues earlier discussed regarding cumulative financial data appearing in Germany’s IATI data - therefore several years of expenditure are tagged on a single date in 2015. The “Germany” IATI data available through d-portal also includes projects not funded through GIZ (e.g. via KfW) so the numbers would not be expected to be identical – though the IATI data does allow the data to be distinguished by implementing partner. JICA publishes quite limited information to IATI; the latest data published to IATI stems from December 2014, according to the IATI Dashboard (it my also refer to an earlier period of spending).

The reasons for the differences for the World Bank and USAID are less clear. For USAID, it could perhaps be an issue to do with spending reported to the AIMS for FY15 that actually occurred in FY14, but may also be due to a large amount non-AIMS-reportable spending from USAID in its IATI data. For the World Bank, the data is being published relatively infrequently, one quarter in arrears, with spending aggregated into six-month periods – so there could be an issue with spending from FY14 either not yet being reported or being attributed to the wrong fiscal year. We understand that there may also be an issue to do with the completeness of the World Bank’s IATI reporting which is currently being addressed.

Below USD 50 million, there are still a number of DPs with large differences, but the numbers are more in line with what would be expected given differences between what is published in IATI data and what can be reported to the AIMS.

#### Chart 2. Comparison of actual disbursements recorded for FY14



On this chart, we only show the difference in disbursement values for FY14 for space considerations. In general, there does seem to be a greater value reported to IATI than to the AIMS. However, it becomes more difficult to identify reasons for any discrepancies when looking across DPs at these smaller values. Some of these differences may also be due to the same spending being reported by different organisations at different stages of the project cycle.

The foregoing discussion underscores the importance of taking a DP-by-DP approach to importing IATI data to the AIMS, to ensure that discrepancies are identified, understood, and can be handled with care. It also shows the benefits of being able to see the numbers that are flowing direct out of headquarters systems – so IATI data can be used as a “sense check” even just by looking at the aggregate numbers.

We do not go into any further comparison of the quality of data available in different systems (e.g. detail of sector coding or geocoding); that is beyond the scope of this report. But it may be useful to conduct this kind of analysis in the later stages of this work in order to help users understand the nature of the data they are importing. It may also be useful as a way of evaluating the relative costs and benefits of using IATI data to supplement data captured by AIMS.

### Surrounding processes

The AIMS is hosted by the Development Effectiveness Wing within the Economic Relations Division (ERD) of the Ministry of Finance. Within ERD, other data is also collected by the Foreign Aid Budget and Accounts (FABA) Unit, which is responsible for debt management and budgeting of foreign aid. The intention is that over time, the AIMS should provide all the data required by FABA and that there should be no need for a parallel data collection process.

We are keenly aware of the need to ensure the data provided by the AIMS to FABA satisfies the needs of FABA – which will be important if the link between the AIMS and FABA is to be strengthened. From our initial meetings, the key priority appears to be to ensure that there is detailed financial data (including actual rather than aggregated financials, because of the need for precise currency conversion), clarity around whether the funds are spent through government systems and a detailed mapping to government budget classifications (which is currently performed manually).

### Technology used in the AIMS

The AIMS uses a combination of Microsoft technology. It is hosted on a Windows server physically located inside the Aid Effectiveness Unit. The database is run on Microsoft SQL Server and the software runs on a .NET MVC framework with C# for the business logic. The front-end uses ASP.net and JQuery for some of the user interfaces. Charts are run through Highcharts and SAP reports are used in a few cases to create printable reports.

### References

### Overview and core resources

* [Use of IATI data in country systems](https://sites.google.com/site/useofiatidataincountrysystems/)
* [International Aid Transparency Initiative](http://aidtransparency.net/)
* [IATI Standard](http://iatistandard.org/) (Contains technical documentation on the IATI Standard, the data format for IATI data)
* [IATI Registry](http://iatiregistry.org/) (Contains links to raw source data published in IATI format.)
* [IATI Datastore](http://datastore.iatistandard.org/) (Queryable datastore for accessing collated data in IATI format.)
* [D-Portal](http://d-portal.org/) (Portal for accessing data in IATI format, by country.)
* [IATI Dashboard](http://dashboard.iatistandard.org/) (Publishing statistics about IATI data)

### Pilots to date

* Development Gateway [Working Paper on IATI and Country Systems](http://www.developmentgateway.org/2015/05/21/iati-and-country-systems-dg-working-paper/)
* [Presentation](http://www.aidtransparency.net/wp-content/uploads/2014/10/DAD-Rwanda-IATI-Integration-14102014-DB-comments.pptx) on pilot with Synergy’s DAD in Rwanda
* [Presentation](http://www.aidtransparency.net/wp-content/uploads/2014/10/PGAI-IATI-Copenhague.ppt) on pilot with Development Gateway’s AMP in DRC

### Useful code

* [IATI](http://github.com/iati) (including the IATI datastore)
* [D-Portal](https://github.com/devinit/D-Portal)
* [Publish What You Fund](http://github.com/pwyf)
* [Development Gateway’s IATI import](https://github.com/devgateway/iatiimport)
* [Aid-budget mapper](http://github.com/markbrough/aid-budget-mapper)
* [OIPA](https://github.com/openaid-IATI/OIPA) (similar to IATI Datastore)

# Data

In Bangladesh, IATI data is of mixed quality. Some DPs reporting to the AIMS have not begun publishing to IATI, while others are publishing only very old data to IATI. On the other hand, there are many DPs with good quality data in IATI, and some DPs publishing to IATI are not accounted for at all in the AIMS.

Publish What You Fund’s [Aid Transparency Index](http://ati.publishwhatyoufund.org/) provides a reasonable proxy for IATI data quality, given that IATI is the most highly-weighted component of the Index. Rather than repeating all of the analysis required for the Index, we use the most recent results across all organisations as a guide.

| **Development Partner** | **FY14 Disbursements (USD millions)** | **FY14 Rank** | **PWYF Rating** |
| --- | --- | --- | --- |
| World Bank | 942.96 | 1 | Very good |
| AsDB | 704.32 | 2 | Very good |
| Japan, JICA | 375.75 | 3 | Poor |
| UK, DFID | 225.32 | 4 | Very good |
| Netherlands | 73.54 | 5 | Fair |
| Australia, AusAID | 45.40 | 6 | Fair |
| EC | 44.91 | 7 | Good |
| UNICEF | 43.48 | 8 | Good |
| GFATM | 39.17 | 9 | Good |
| Canada, DFATD | 34.36 | 10 | Good |
| UNFPA | 32.44 | 11 | Not rated |
| WFP | 30.45 | 12 | Not rated |
| Switzerland, SDC | 28.23 | 13 | Fair |
| IFAD | 27.92 | 14 | Not rated |
| Germany, GIZ | 19.49 | 15 | Fair |
| UNDP | 11.47 | 16 | Very good |
| Korea, KOICA | 10.29 | 17 | Poor |
| OFID | 10.11 | 18 | Not rated |
| Denmark, Danida | 8.11 | 19 | Fair |
| Sweden, Sida | 7.67 | 20 | Very good |

Disbursements refer to AIMS data as at 2015-11-17 for FY14, in millions of USD. PWYF Rating refers to Publish What You Fund’s 2014 Aid Transparency Index

## Analysis of specific donors’ data

A number of DPs face specific challenges with their data. These challenges are captured in this section not to criticise them, but to ensure that they are adequately taken into account in the course of this work. In some cases this may mean DPs changing the way they publish data at headquarters level, and in others it may mean that the importing tool needs to be flexible to deal with nuances in the way data is published. In time, there may be arguments for changes to the IATI Standard or the way in which it is interpreted in order to make it easier for software to handle these specificities or differences. However, we suggest any changes to the Standard should be made only once different approaches have been thoroughly tested.

## Canada

Canada has a much larger number of projects in its IATI data than in its AIMS data. Some of these projects may have been excluded for historical or other reasons. However, in working out how best to represent projects that are co-funded / trust funds, etc, it is interesting to look at similar projects that have been excluded.

### Considering how to handle trust funds etc.

[This dataset](https://github.com/BD-IATI/donor-data/raw/aae21ba2196d1b1b85be1c2d938b5510d3b58d81/canada/iati_projects_and_aims.xlsx) contains the full list of Canadian projects in IATI data, filtered for projects of aid type B03 or B04(contributions to multilaterals), and excluding projects that are tagged as Bangladesh for 20% or less (as opposed to other countries). Where projects are marked NOT FOUND on the left, the project was not found in the IATI data.

Focusing on those projects where the implementing organisation is an organisation that also reports to the AIMS (UNFPA, UNICEF and World Bank):

* projects in **green** were not found in the AIMS (and are presumably reported by the implementing organisation)
* projects in **orange** were found in the AIMS (and are presumably not reported by the implementing organisation)

It will be important to work out what features of a project mean that it should be reported by the funder vs the implementer in this context. We also need to work out how to identify when an activity needs to be mapped to another organisation’s activity when the two relate to the same project. We also need to work out how to identify projects that are contributions to (it seems exclusively World Bank-adminstered) Trust Funds.

### Example project

e.g. Project CA-3-A035529001 (“Skills Training and Enhancement Project”) [[d-portal](http://www.d-portal.org/ctrack.html?country=BD#view=act&aid=CA-3-A035529001)] [[IATI Datastore](http://datastore.iatistandard.org/api/1/access/activity.xml?iati-identifier=CA-3-A035529001)]

* this is reported in the AIMS as a [World Bank project P090807](http://aims.erd.gov.bd/AIMS/ProjectInfo/Details/1188);
* the money comes from a [Trust Fund TF015113](http://aims.erd.gov.bd/AIMS/TrustFund/Index) with 100% of the funds coming from Canada (NB, that link can only be viewed by users with certain privileges);
* it’s then implemented by ERD and the Ministry of Education.

## DFID

DFID publishes projects and sub-components in its IATI data. The approvals process in Bangladesh means that projects can only be reported after a certain stage, so certain of DFID’s project components (preparatory work, M&E) should not be reported to the AIMS. The import interface will need to allow components to be deselected from projects before importing. It is possible that other DPs will face a similar challenge, and they may have a less clear division in their data.

All DFID AIMS projects appear to be found in IATI data. There are a significant number of projects in DFID’s IATI data that are not found in the AIMS.

### Project components

Question: how does DFID enter projects into the AIMS when:

1. component A is a trust fund contribution
2. component B is funded through another organisation

… assuming both components A and B are includable in the AIMS / approved? Is it only the value of component B that should be included?

See also comparison of DFID projects reported in AIMS vs IATI:

* [comparison of DFID projects](https://github.com/BD-IATI/donor-data/blob/master/dfid/dfid.ipynb) found in IATI vs the AIMS – scroll down on that page for the comparison of projects
* [download the comparison table here](https://raw.githubusercontent.com/BD-IATI/donor-data/master/dfid/iati_projects_and_aims.xlsx)

### Implementing organisations

DFID appears to be using [CRS channel codes](http://iatistandard.org/201/codelists/CRSChannelCode/) for implementing organisations. These include some broad categories like 11000 (“donor”) and 50000 (“other”). However, significantly more information about organisations is provided in each transaction’s receiver-org.

* These two pieces of information together might provide us with enough information about which other organisation the project should map to.
* However, it is probably not desirable for the import tool to have too many donor-specific “hacks”.
* In some cases there are multiple receiver orgs, but in these cases there is generally a dominant receiver org by value.
* There is also an issue where the specific organisation is in quite a few cases not stated, though it may be obvious from the title or description of the activity which organisation it is (so this has nothing to do with security exclusions etc.)

See comparison of DFID implementing organisations and receiver organisations:

* [comparison of DFID implementing orgs and receiver orgs](https://github.com/BD-IATI/donor-data/blob/2d267cd026343f177addde1c6fc7ac9a1d468c9d/dfid/dfid_transactions_implementers.xlsx) - annotated cells (in red) shows where:
  + the project is probably implemented by an organisation reporting elsewhere in the AIMS
  + it is unclear which organisation is referred to
* [flat version of the same file](https://github.com/BD-IATI/donor-data/raw/master/dfid/dfid_transactions_implementers_flat.xlsx) - to facilitate further analysis

## Germany

Germany’s data is published by BMZ and includes projects funded through GIZ and KfW. The data currently includes cumulative figures for each project, without any breakdown over time. This makes it difficult to know how much has been spent in any year. GIZ projects also contain multiple “phases”, but it is unclear if these phases are or should be joined together to make a single project in the IATI data.

### Project components

A small number of projects captured in the AIMS do not appear in IATI. One project is divided in two in the AIMS but combined in IATI data. The other project does not appear in IATI data; it is unclear why. GIZ’s [project database](https://www.giz.de/projektdaten/index.action#?region=2&countries=BD)contains far more proejcts than those published in IATI, even restricting the search to only those projects financed by BMZ, Germany’s development ministry. Again, it is not clear why.

The [BMZ project portal](http://www.bmz.de/de/ministerium/zahlen_fakten/transparenz-fuer-mehr-Wirksamkeit/iati/index.jsp) appears to be out of date, but this data also raises an important issue. Where GIZ projects are comprised of numerous phases, the project number of only the most recent phase is published. It is not clear if this is because older phases are deleted, or if the phases are combined and labelled with the most recent phase’s project ID. Both options are highly undesirable because it means either that there is missing data or that project IDs are not stable.

See also comparison of Germany projects reported in AIMS vs IATI:

* [comparison of German projects](https://github.com/BD-IATI/donor-data/blob/master/germany/germany.ipynb) found in IATI vs the AIMS – scroll down on that page for the comparison of projects
* [download the comparison table here](https://raw.githubusercontent.com/BD-IATI/donor-data/master/germany/iati_projects_and_aims.csv)

## World Bank

The World Bank administers a number of trust funds. It will be important to work out how to handle these trust funds in the data.

Our understanding is that trust funds are currently not published in the World Bank’s IATI data. The World Bank IATI data also contains financial data only one quarter (or more) in arrears - the latest data (as of 2016-01-27) was dated from 2015-09-30. This is too old to be useful on an ongoing basis. Disbursement data is also aggregated by quarter; a minimum of monthly disaggregation would be preferable.

*See also comparison of World Bank projects reported in AIMS vs IATI:*

* [comparison of World Bank projects](https://github.com/BD-IATI/donor-data/blob/master/worldbank/worldbank.ipynb) found in IATI vs the AIMS – scroll down on that page for the comparison of projects
* [download the comparison table here](https://raw.githubusercontent.com/BD-IATI/donor-data/master/worldbank/iati_projects_and_aims.csv)

## Other donors

### Netherlands

For the the Netherlands’ projects, commitments are made in the local currency value - in Bangladesh, this is USD rather than BDT. The commitment value is stored in the Ministry of Foreign Affairs’ internal project management system in EUR and USD. The EUR value is shown in IATI data. At the start of each year, the EUR commitment values are recalculated to ensure that the local project value in USD is accurately reflected in EUR, given currency fluctuations over the previous year. This should be fairly straightforward to handle given the mechanisms the IATI Standard has for dealing with currencies, but it is useful to be aware of this nuance. The Netherlands also spends much of its funds through large tranches of programmatic funding to implementing partners such as Dutch NGOs. Including IATI data published by Dutch NGOs may help to provide a fuller picture of Dutch development cooperation projects.

### UNDP

UNDP receives funds from headquarters and from DPs in country to fund particular projects. It is important to avoid double-counting the DP-reported projects with the UNDP-reported projects. We propose to handle this issue by allowing DPs to map their projects to the UNDP project, and then allow UNDP to decide which data it chooses to take - its own, or others’ data. This issue of data being reported at multiple levels will certainly apply to other organisations and the user interface will need to handle it in a general way.

### UNICEF

UNICEF has a unit of aid in its IATI data that shows results rather than projects. This issue cannot be solved in the user interface but would need to be taken up with headquarters to work out if the system could instead export project data.

### USAID

USAID has a very granular unit of aid in its IATI data (“awards”), so activities will need to be grouped together to create meaningful projects. It appears that in Bangladesh, one project has one or many associated awards, so an interface to allow grouping of awards into projects will be useful. There may however be awards belonging to multiple projects - we propose that such awards should not be “split” for now, but should be handled manually. Systems improvements currently underway at USAID could in time help to improve this process

# Development

## Development process

The development process will need to be highly agile and iterative given the necessity to experiment with different approaches and techniques to handling the data, as well as responding to user feedback to ensure the end product is user friendly and intuitive. Some parts of the development process can happen in parallel – particularly back-end and front-end work – so that we can proceed more quickly as well as benefit from the specialisations of individual developers.

We will provide mock-ups of the front-end interface and work closely with the developers to fully understand what is being developed and how it relates to the data. All code will need to be on Github from the start, and in a public repository, so that we can benefit from collaboration with others (who also have their work on Github) as well as ensure that the final product will be useful in a wider range of environments and more likely to be maintained.

We will leverage existing tools and care strongly about maintenance and sustainability – so it may be preferable to use similar languages used in the rest of the IATI community (notably Python).

## Indicative development roadmap

In the **first phase**, data will be extracted from IATI and the AIMS, with projects then listed side by side on the same page. This will include a field-by-field cross-walk and mapping. Requires AIMS to export data in IATI-XML format for each donor.

In the **second phase**, donors will map their projects in IATI to those already contained in the AIMS. Establishing this project-level relationship is arguably the most important part to get right and we emphasise the challenge of this component - both in technical and conceptual terms.

The **third phase** will then test the import of a limited subset of fields - probably restricted to the title and description - into the AIMS. Requires AIMS to accept data from the IATI import module.

In the **fourth phase**, we will begin to establish techniques to reconcile multi-donor projects, focusing on co-financed projects.

In the **fifth phase**, automatically updating fields with data from the IATI Registry will be established. This will need to provide a user-friendly interface for controlling updates, as well as development of logic for handling different data sources. Requires, inter alia, AIMS to know whether a field comes from IATI or has been manually entered.

In the **sixth phase**, we will move to encompass other fields - particularly financial data (transactions, including disbursements).

In the **seventh phase**, we will develop techniques required to handle multi-donor projects (parallel financing, co-financed projects, trust funds, etc.) and tools to mitigate against double-counting.

Again, various aspects of the seven phases can be carried out in parallel. The implementation will be determined as a success if donors with high-quality IATI data are able to import data into the AIMS. Due to the short timeframe, the amount of time available for the later phases may be limited.

## Sprint 1

**STATUS: CONFIRMED**

* [Overview](http://bd-iati.github.io/development/sprint-1/#overview)
  + [Data integrity](http://bd-iati.github.io/development/sprint-1/#data-integrity)
  + [DP data overview](http://bd-iati.github.io/development/sprint-1/#dp-data-overview)
* [1. Retrieve data from IATI Datastore](http://bd-iati.github.io/development/sprint-1/#retrieve-data-from-iati-datastore)
  + [Organisations](http://bd-iati.github.io/development/sprint-1/#organisations)
* [2. Conversion](http://bd-iati.github.io/development/sprint-1/#conversion)
  + [Simplifying conversion](http://bd-iati.github.io/development/sprint-1/#simplifying-conversion)
* [3. Storage](http://bd-iati.github.io/development/sprint-1/#storage)
  + [Storing individual activities](http://bd-iati.github.io/development/sprint-1/#storing-individual-activities)
* [4. Make data available to rest of application](http://bd-iati.github.io/development/sprint-1/#make-data-available-to-rest-of-application)
* [5. Read data from the AIMS in the IATI Standard](http://bd-iati.github.io/development/sprint-1/#read-data-from-the-aims-in-the-iati-standard)
* [6. Write data to the AIMS](http://bd-iati.github.io/development/sprint-1/#write-data-to-the-aims)

### Overview

The main goals of sprint 1 are:

1. download IATI data from the IATI Datastore for individual DPs
2. convert between versions of the IATI Standard
3. store the data
4. make it accessible to the rest of the application
5. read data from the AIMS in the IATI Standard
6. write data to the AIMS

As this is the first sprint, this does not all have to be working perfectly. The most important thing is to get the basic framework up and running and to establish a decent architecture for the application. This sprint also needs to function nicely with the user interface, which **will be developed in parallel** under sprint 2.

### Data integrity

* data in the live AIMS should not be touched
* data in the test AIMS can be changed and adjusted as necessary for the purposes of testing
* data in the IATI import tool can be deleted and re-generated at will throughout the development process, until we agree otherwise, perhaps in some of the last few sprints.

### DP data overview

For the purposes of this sprint, we will use data from Canada and DFID.

|  | **Canada** | **DFID** |
| --- | --- | --- |
| Organisation identifier \* | CA-3 | GB-1 |
| Version | 2.01 | 1.05 |
| Hierarchies | 1 | 2 |
| Languages | English, French | English |

\* this is currently the only- or most-used identifier. There may eventually be more than one identifier in use; see below.

The data from both DPs is good. See also some more detailed data analysis for [Canada](http://bd-iati.github.io/data/canada/) and [DFID](http://bd-iati.github.io/data/dfid/).

### 1. Retrieve data from IATI Datastore

Data should be downloaded nightly from the IATI Datastore and then made available to the rest of the application on demand. See the documentation section [Retrieving IATI data](http://bd-iati.github.io/documentation/retrieving-data/) for more detail on the query for retrieving data from the IATI Datastore.

#### Organisations

* The list of Development Partners can be obtained from the AIMS, using the data collected in [FundSource](http://aims.erd.gov.bd/AIMS/FundSource).
* The JSON endpoint used in the AIMS could be used, or some other method for supplying data to the IATI import module could also be considered. The software supplier are free to decide what is most appropriate.
* **Mark** will fill in the IATI organisation identifier for several organisations.
* Not all organisations will have organisation identifiers supplied, because they have not published IATI data yet.
* Organisations could theoretically have multiple organisation identifiers. As there is a single field in the AIMS for organisation identifiers, they will be delimited with a | character. This is the same character that is used as an OR operator in the IATI Datastore. So, the reporting-org parameter passed to the Datastore API would be:
* reporting-org=GB-1|XM-DAC-12-1

where GB-1 and XM-DAC-12-1 are both organisation identifiers for DFID.

### 2. Conversion

There are multiple versions of the IATI Standard. See the documentation section [Versions in IATI data](http://bd-iati.github.io/documentation/versions/) for an overview of the key differences between versions.

We will convert all the data to version 2.02 for use in this application. For our purposes, there are no relevant differences between version 2.01 and version 2.02.

In the initial sprint, we need to convert version 1.05 data to version 2.02. This should be developed in a separate module from the outset if possible, as it will likely be very useful for others trying to use IATI data.

#### Simplifying conversion

We will take some steps to simplify the process of conversion for now. We can ignore the sub-national geographic location changes between version 1.03 and 1.04 (we will need to have this conversion working by the end of the project).

As outlined in the documentation page, the main changes between v1.05 and v2.02 are:

1. several codelists were changed from English-language codes to numeric codes
2. changes to improve the way multiple languages are handled in the Standard

We should focus initially on adjusting the following fields in each of these two main changes:

1. codelist changs – swap the codes from v1.05 to v2.02 in the following fields:
   * activity-date/@type: [activity date type](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#activity-date-type-amended-codes)
   * participating-org/@role: [organisation role](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#organisation-role-amended-codes)
   * sector/@vocabulary: [sector vocabulary](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#sector-vocabulary-was-vocabulary-amended-codes)
   * transaction/transaction-type/@code: [transaction type](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#transaction-type-amended-codes)
2. multiple languages – combine the following elements:
   * title
   * description (with the same type)
   * sector (with the same code; only where vocabulary is DAC for now)

This should be done in a flexible way so that we can gradually (and fairly painlessly) increase the number of fields we are converting over the course of development.

### 3. Storage

The software supplier is free to decide how to store the data at this stage. It will probably make most sense to have a database that stores information such as:

* DP name
* DP organisation identifier(s) in IATI
* DP organisation identifier in AIMS (as appropriate)
* Date/time data was last downloaded

The XML files themselves could be stored in the database and/or as flat files.

Note that we need to have some way of storing or accessing the AIMS data. This could be “on the fly”, passed through directly from reading the AIMS database, or activities could be written to some table in the database.

#### Storing individual activities

As we approach the later stages of development and want to re-sync the data with the Datastore (to fetch new activities, or to update existing ones), we’ll need to know what the activity looked like the last time we saw it. It might make sense to store each activity in a separate row in the table, like this:

| **key** | **value** | **description** |
| --- | --- | --- |
| organisation\_id | GB-1 | Reporting organisation’s IATI identifier |
| iati\_identifier | GB-1-107368 | Activity’s IATI identifier |
| last\_downloaded | 2016-02-08Z19:00:00 | The timestamp the file was most recently downloaded |
| previous\_downloaded | 2016-02-07Z19:00:00 | The timestamp the file was previously downloaded |
| last\_xml | <iati-activity ...> | The XML blob of the activity, for the most recent file |
| previous\_xml | <iati-activity ...> | The XML blob of the activity, for the previous file |
| hierarchy | 1 | This activity is a 1st-level activity in the structure DFID uses |
| parent\_hierarchy | null | There are no parents of this activity (this would be used to define the parent of any hierarchy-2 (etc) activities) |

We may want to store the original XML and the converted XML here for the purposes of debugging.

There’s probably no need right now to store the data in a more complicated way than this (e.g. splitting out titles into a separate table, etc). But we may want to revisit this later when we move on to re-syncing and tracking changes and updates in files.

### 4. Make data available to rest of application

This is going to be one of the trickier parts of this sprint and the part that may need to change somewhat as development progresses. We need to have some kind of API to read data from the database and make it available to the front end.

This could either be the full data about each activity, or it could provide more limited data at this stage. Given that we want to show only limited information to the user at this stage, the latter option is probably preferable. This will also reduce the amount of data that is read out of the system and make it easier / faster to have more of the interface in the browser, if we decide to go down that path in sprint 2.

A JSON endpoint for providing this data probably makes sense.

At a minimum, we need the following fields to be made available about each activity:

* iati-identifier
* title (English title, xml:lang="en", if multiple languages provided)
* description (again, in English)
* aid type
* implementing organisation name
* implementing organisation ref
* implementing organisation type
* recipient country (% tagged as BD - Bangladesh)

A couple of notes on accessing this data:

* **aid type** can be declared once at the activity level (default-aid-type) or multiple times at the transaction level (aid-type). For now, we can take either, and return one. We may later want to consider what to do if multiple different aid types appear, but we will leave this complication to one side for now.
* **implementing organisation** is generally declared using the role of participating-org. However, we should also consider combining this with the receiver-org element on each transaction. Again, we will simplify this for now, but will eventually want to have some logic that can look at either. In the case of the transactions, we could select the receiver-org with the highest value of disbursements.

Finally: this endpoint should make both IATI and AIMS data available in the same format (and a similar endpoint) so that it is easy to pull together in the user interface.

### 5. Read data from the AIMS in the IATI Standard

The AIMS already publishes data in XML. The export should be adjusted so that:

* the XML is compliant with version 2.02 of the Standard
* an endpoint to obtain all activities from each DP
* an endpoint to obtain all activities from all DPs
* both endpoints are publicly accessible, but not necessarily publicly discoverable.

Eventually, we may want to make the “all data” endpoint publicly discoverable. At this point, there may be value in cacheing the output to reduce server load. However, we would want the IATI import module to see the “live” data rather than cached data.

In addition to the above data, we need to retrieve some additional metadata (which doesn’t have to be in XML format):

* [Fund Source](http://aims.erd.gov.bd/AIMS/FundSource)
* [Trust Funds](http://aims.erd.gov.bd/AIMS/TrustFund) (but NB, data needs to include breakdown of commitments from different organisations that are shown in each of those pop-ups)

### 6. Write data to the AIMS

At this stage, we only need a simple API that can:

* write new activities to the AIMS for DPs already listed in [FundSource](http://aims.erd.gov.bd/AIMS/FundSource)
* update activities in the AIMS

This does not need to be all fields initially. We should start with titles and descriptions.

## Sprint 2

**STATUS: CONFIRMED**

* [Overview](http://bd-iati.github.io/development/sprint-2/#overview)
  + [Data integrity](http://bd-iati.github.io/development/sprint-2/#data-integrity)
  + [DP data overview](http://bd-iati.github.io/development/sprint-2/#dp-data-overview)
* [1. Build a basic interface](http://bd-iati.github.io/development/sprint-2/#build-a-basic-interface)
* [2. Filter and group projects](http://bd-iati.github.io/development/sprint-2/#filter-and-group-projects)
  + [a. Establish hierarchy](http://bd-iati.github.io/development/sprint-2/#a-establish-hierarchy)
  + [b. Filter out projects likely not relevant to Bangladesh](http://bd-iati.github.io/development/sprint-2/#b-filter-out-projects-likely-not-relevant-to-bangladesh)
  + [c. Determine whether projects should be reported by this DP or another](http://bd-iati.github.io/development/sprint-2/#c-determine-whether-projects-should-be-reported-by-this-dp-or-another)
* [3. Match projects between IATI and AIMS](http://bd-iati.github.io/development/sprint-2/#match-projects-between-iati-and-aims)
* [4. Display matched and unmatched projects](http://bd-iati.github.io/development/sprint-2/#display-matched-and-unmatched-projects)

### Overview

The main goals of sprint 2 are:

1. build a basic interface that can read data from the API defined in sprint 1
2. ask the user a series of questions to try to filter and group projects into a) those that this user is responsible for reporting and b) those that other users are responsible for reporting
3. match projects in IATI data to those in the AIMS; and obtain a list of IATI and AIMS projects that do not match;
4. display related IATI and AIMS projects side by side.

### Data integrity

* data in the live AIMS should not be touched
* data in the test AIMS can be changed and adjusted as necessary for the purposes of testing
* data in the IATI import tool can be deleted and re-generated at will throughout the development process, until we agree otherwise, perhaps in some of the last few sprints.

### DP data overview

For the purposes of this sprint, we will use data from Canada and DFID.

|  | **Canada** | **DFID** |
| --- | --- | --- |
| Organisation identifier \* | CA-3 | GB-1 |
| Version | 2.01 | 1.05 |
| Hierarchies | 1 | 2 |
| Languages | English, French | English |

\* this is currently the only- or most-used identifier. There may eventually be more than one identifier in use; see below.

The data from both DPs is good. See also some more detailed data analysis for [Canada](http://bd-iati.github.io/data/canada/) and [DFID](http://bd-iati.github.io/data/dfid/).

### 1. Build a basic interface

It is up to the software supplier to design / suggest a basic interface. Mock-ups will be provided at<http://test.brough.io/bd/> – but these will be some ideas and suggestions; the software supplier does not have to follow them rigidly and is encouraged to suggest alternatives.

The interface needs to be flexible enough to allow data for a particular donor to be filtered and mapped to the AIMS data, including (eventually):

* mapping individual projects in IATI to projects in the AIMS
* providing some granularity on field-level mapping between IATI and AIMS projects
* allowing for different [types of projects](http://bd-iati.github.io/documentation/project-types/) to be handled in different ways
* providing user login and authentication
* being fast and responsive in low-bandwidth environments

This user interface can be gradually be built up and improved upon over the course of the sprints. It can potentially be quite plain at the outset (e.g. undesigned wireframes), if there is a need to concentrate on the other components of the sprint. However, it should eventually look clean and slick on most modern browsers (IE 7+ unless particular desired libraries have compatibility issues – we can adjust this requirement by mutual agreement).

### 2. Filter and group projects

#### a. Establish hierarchy

Some DPs will use a hierarchy of projects in their data. We need to ask the DP which hierarchy they would like to map to the AIMS. We should provide them with our recommendation, generated by establishing the % of activities at each hierarchy that have a match to the AIMS. The highest match wins.

We should also show some sample data to the user (iati-identifiers and titles for one parent activity and all of its chidren, to a maxmimum of 5 activities at each level), so that they can understand the decision they are making. This can look something like [this example](http://bd-iati.github.io/documentation/hierarchies/#understanding-hierarchies-in-iati).

#### b. Filter out projects likely not relevant to Bangladesh

Show all activities on a page. Deselect those where:

* recipient-country percentage for Bangladesh is less than 20%
* aid type is B01 (core support to NGOs) or B02 (core contributions to multilaterals)
* activity-status is not 2 (i.e. closed projects)

We will want to revisit these filters later, so they should be implemented in a flexible way.

The user should be able to move projects from one section to another (i.e. to manually select to include or exclude particular projects)

#### c. Determine whether projects should be reported by this DP or another

IATI data may include multiple activities for the same project, as each organisation involved can publish the part they are responsible for. For example, DFID may publish its contribution to a World Bank project as one activity, and the World Bank may publish its implementation of that project as another activity.

Generally, where one DP A is giving funds to another DP B registered elsewhere in the AIMS, DP B should be responsible for the activity, and DP A should be listed a providing contributions to that activity.

We therefore need to try and determine if a DP’s activities are likely relating to their own activities (“mine”) or another DPs (“not mine”). See also a brief discussion of [project types](http://bd-iati.github.io/documentation/project-types/) in the documentation.

In a similar way to the previous step (b), we want to filter out projects, but in a flexible way so that the DP can choose to adjust the results if they do not agree.

Projects are determined as “not mine” if:

* the implementing organisation is of organisation type 40 (“multilateral”);
* the implementing organisation has a ref 11000 (“donor governments”) or 13000 (“third-country governments, delegated cooperation”)
* the implementing organisation has a ref that matches a “fund source” from the AIMS – NB there is a field to state the IATI identifier of the organisation, but will not always be filled out

The user should be presented with a list of unique implementing organisations, and asked to determine if any of those are known to exist elsewhere in the AIMS. For example, the implementing organisation could be listed as simply “WB” which the user can easily identify as “World Bank”.

There should be a list of projects in each section and it should be possible to manually move projects from one section to another.

### 3. Match projects between IATI and AIMS

Initially, to match projects between IATI and the AIMS, we will rely on the project ID or IATI Identifier being filled out in the AIMS.

We can check this two ways:

* IATI’s iati-identifier matches the “IATI Identifier” field in the AIMS
* IATI’s iati-identifier, excluding the organisation identifier (found in reporting-org), matches the “IATI Identifier” or “Project ID” field in the AIMS

An overview of the project IDs in the AIMS suggests this data is fairly good, so this should work well. We can revisit alternate options for trying to match projects if this does not work sufficiently well.

### 4. Display matched and unmatched projects

This page should have four sections:

1. Projects that matched between IATI and the AIMS
2. IATI projects that were not found in the AIMS
3. AIMS projects not found in IATI
4. Projects defined as “not mine”

In the first two sections, only projects defined as “mine” should be shown.

## Sprint 3

**STATUS: CONFIRMED**

* [Overview](http://bd-iati.github.io/development/sprint-3/#overview)
  + [Data integrity](http://bd-iati.github.io/development/sprint-3/#data-integrity)
  + [DP data overview](http://bd-iati.github.io/development/sprint-3/#dp-data-overview)
* [1. Build a general field selection interface](http://bd-iati.github.io/development/sprint-3/#build-a-general-field-selection-interface)
  + [Storing preferences](http://bd-iati.github.io/development/sprint-3/#storing-preferences)
* [2. Display a list of projects](http://bd-iati.github.io/development/sprint-3/#display-a-list-of-projects)
* [3. Project-level field preference selection](http://bd-iati.github.io/development/sprint-3/#project-level-field-preference-selection)
* [4. Test import of basic data fields to the test AIMS system](http://bd-iati.github.io/development/sprint-3/#test-import-of-basic-data-fields-to-the-test-aims-system)
* [5. Think about how to handle financial data](http://bd-iati.github.io/development/sprint-3/#think-about-how-to-handle-financial-data)

### Overview

The main goals of sprint 3 are:

1. create a general field selection interface - providing a template for where to generally take data from for this donor’s projects - and store those preferences for next time
2. display a list of projects with a couple of warning flags where the data looks strange
3. allow a user to override these “general preferences” for specific projects
4. test import of basic data fields to the test AIMS system (titles and descriptions)
5. think a little about how we will handle financial data.

### Data integrity

* data in the live AIMS should not be touched
* data in the test AIMS can be changed and adjusted as necessary for the purposes of testing
* data in the IATI import tool can be deleted and re-generated at will throughout the development process, until we agree otherwise, perhaps in some of the last few sprints.

### DP data overview

For the purposes of this sprint, we will use data from Canada, DFID, and the European Union (EU only if we have time).

|  | **Canada** | **DFID** | **EU** |
| --- | --- | --- | --- |
| Organisation identifier \* | CA-3 | GB-1 | XI-IATI-EC\_DEVCO |
| Version | 2.01 | 1.05 | 1.04 |
| Hierarchies | 1 | 2 | 2 |
| Languages | English, French | English | English |

\* this is currently the only- or most-used identifier. There may eventually be more than one identifier in use; see below.

The data from both DPs is good. See also some more detailed data analysis for [Canada](http://bd-iati.github.io/data/canada/) and [DFID](http://bd-iati.github.io/data/dfid/).

### 1. Build a general field selection interface

A DP should see all projects that they manage. In other words, activities that at the end of sprint 2 have been determined as not managed by any other DP. DFID would see their activities and not any activities that they have delegated to another DP such as the World Bank.

We need to have a way of allowing the user to provide some general preferences about where data should be sourced from for each field. Our assumption is that data quality is likely to be fairly consistent within individual DPs’ data – so if descriptions are poor for one or two projects, they’re likely to be poor for the rest of that DP’s data.

For these purposes, we should select **one project** from IATI and the same project from the AIMS. We need to show the project data side by side and give the user the opportunity to select where they would prefer to take the data from.

Right now, we only care about being able to import titles and descriptions, but we should also build a basic interface to show financial data from either system.

### Storing preferences

Once a user has provided their general import preferences, we should store these so that we can present the same options back to them next time (for example, when a new project comes along). This could be in a really simple table like the following:

| **DP** | **Field** | **Source** |
| --- | --- | --- |
| DFID | title | AIMS |
| DFID | description | IATI |
| DFID | transaction | IATI |

I think it probably makes sense to just use the element name (e.g. title) as the name of the field. I can’t see a reason why we would want any more granular controls than that.

The interface should read from this preferences table when loading the interface, and display the user’s previous choices. This means that in future, when new projects are published in their IATI data, they can choose to import them easily using the same set of preferences that they previously stated.

[see mockup, tab “6. Set import preferences”](http://test.brough.io/bd/sprint3.htm)

### 2. Display a list of projects

We should then display a list of all the projects and show some summary financial data about each project, highlighting those where there are big discrepancies. If we have enough room and it isn’t too cluttered, ideally we would show both commitments and disbursements. We could then highlight those areas where commitments or disbursements vary by over 5%.

We may want to think about other heuristics we could use to show other warning flags at this stage, but this is probably sufficient for now.

[see mockup, tab “7. Review and adjust before import”](http://test.brough.io/bd/sprint3.htm)

### 3. Project-level field preference selection

Having specified the general preferences on field selection, we also want to allow the user to override this for specific projects if necessary. The interface should look the same as the general preferences interface.

The mockup for the previous step suggests that we allow a user to adjust preferences for individual projects by clicking on a little “edit” button next to the project. That probably makes sense, though we should probably also make it possible to scroll through projects (see field selection for the next project, for example).

### 4. Test import of basic data fields to the test AIMS system

Once the user has established their preferences for field import, we can then proceed with importing those fields to the **test** AIMS system. For the purposes of this sprint, we will only import the **title** and **description** from each project.

The only mockup we have for this step is the button at the bottom of the “Review and adjust before import” step. We should also have some basic information provided back to the user about whether the import process was successful or not.

[see mockup, tab “7. Review and adjust before import”](http://test.brough.io/bd/sprint3.htm)

### 5. Think about how to handle financial data

In the proceeding steps, we don’t need to try and import any financial data. However, we should have a think about how we present the data and how we might deal with it later.

In this sprint, we should at least be able to:

* present the total value of commitments and disbursements respectively, for each project, in each system
* show the total values of commitments and disbursements in each system

At some point, we should also try to:

* show the total value of commitments / disbursements by year
* show the total value of commitments / disbursements by fiscal year

If we can get some of this working now then that will put us in a good position later.

## Sprint 4

**STATUS: CONFIRMED**

* [Overview](http://bd-iati.github.io/development/sprint-4/#overview)
  + [Data integrity](http://bd-iati.github.io/development/sprint-4/#data-integrity)
  + [DP data overview](http://bd-iati.github.io/development/sprint-4/#dp-data-overview)
* [1. Allow users to pass activities over to each other](http://bd-iati.github.io/development/sprint-4/#allow-users-to-pass-activities-over-to-each-other)
* [2. Development of interfaces to handle more complex projects](http://bd-iati.github.io/development/sprint-4/#development-of-interfaces-to-handle-more-complex-projects)
* [3. Trust Funds “overview interface”](http://bd-iati.github.io/development/sprint-4/#trust-funds-overview-interface)
* [4. Co-financing “overview interface”](http://bd-iati.github.io/development/sprint-4/#co-financing-overview-interface)
* [5. User management](http://bd-iati.github.io/development/sprint-4/#user-management)

### Overview

The main goals of sprint 4 are:

1. allow users to pass activities over to each other
2. development of interfaces to handle more complicated projects
3. Trust Funds simple interface
4. Co-funding simple interface
5. implement some user management (authentication / authorization)

### Data integrity

* data in the live AIMS should not be touched
* data in the test AIMS can be changed and adjusted as necessary for the purposes of testing
* data in the IATI import tool can be deleted and re-generated at will throughout the development process, until we agree otherwise, perhaps in some of the last few sprints.

### DP data overview

For the purposes of this sprint, we will use data from Canada, DFID, UNDP and World Bank.

|  | **Canada** | **DFID** | **UNDP** | **World Bank** |
| --- | --- | --- | --- | --- |
| Organisation identifier \* | CA-3 | GB-1 | XM-DAC-41114 | 44000 |
| Version | 2.01 | 2.01 | 1.04 | 1.05 |
| Hierarchies | 1 | 2 | 2 | 1 |
| Languages | English, French | English | English | English |

\* this is currently the only- or most-used identifier. There may eventually be more than one identifier in use; see below.

The data from all four DPs is good. We’re looking at UNDP and World Bank data in this sprint because they operate projects funded by Canada and DFID. Note that the World Bank does not yet publish Trust Funds in its IATI data.

### 1. Allow users to pass activities over to each other

In sprint 2 (step 3) we allow users to “Filter DP activities” - as a way of sorting activities into:

1. activities that are the DP’s own projects
2. activities that are the DP’s contributions to projects managed by **other DPs**

We now want to save the second set of activities and record them as having been passed across to the other DP. This could just be a question of altering [the basic table we created in sprint 1](http://bd-iati.github.io/development/sprint-1/#storing-individual-activities) by adding an additional two columns:

| **key** | **value** | **description** |
| --- | --- | --- |
| assigned\_to\_organisation\_id | 44001 | This project has been assigned to another organisation (default value should probably be the original organisation’s organisation\_id? Or perhaps null.) |
| assigned\_to\_datetime | 2016-02-29 19:53:00 | A timestamp for when the project was assigned to another organisation. |

We may want to think about a notification system within the app for alerting other users that these sorts of things have happened, though perhaps we can just use the assigned\_to\_datetime to alert the organisation receiving this activity when they log on.

### 2. Development of interfaces to handle more complex projects

There are two main types of complicated projects we need to handle:

1. **Trust Funds** – where different organisations put money into a big pot (the pot is managed by a particular DP - normally the World Bank), and projects are then funded out of that pot;
2. **Co-funding** – where different organisations agree to fund the same project, but disburse their money separately, directly to the implementing organisation.

The main way that we handle this is to create an interface for the organisation that gets assigned the activities (let’s call them the **Managing DP**). It then becomes that organisation’s responsibility to sort out how activities should be mapped together.

The Managing DP, when they log in, should be presented with a list of activities that have been assigned to them, with the option to map them against:

* one of their own projects, **OR**
* a trust fund.

In the drop-down lists in the mock-up below, **all the Managing DP’s projects** and **all trust funds** should be shown.

[see mockup “Activities from other DPs”](http://test.brough.io/bd/sprint4.htm)

### 3. Trust Funds “overview interface”

Once the Managing DP has assigned activities to their own projects or to a Trust Fund, they should be able to decide how to handle this data.

In this interface, we assume that the user has the right to handle Trust Funds. In reality, we may only want to give these permissions to some DPs (e.g. the World Bank) – but we can come back to that later.

We should create a simple overview of the Trust Funds interface that allows the Managing DP to decide what to do with the commitments that have been assigned to them (and which they have then mapped against a Trust Fund). They can choose to allow these new commitments to be entered into the AIMS, or to reject them.

In the AIMS, the cumulative amount of commitments per DP is stored rather than any breakdown over time. In IATI data, we can show individual commitments.

We don’t need to enter this data to the AIMS at this point – just to create the interface that makes it possible.

[see mockup “Trust Funds”](http://test.brough.io/bd/sprint4.htm)

## 4. Co-financing “overview interface”

Once the Managing DP has assigned activities to their own projects, they should be able to decide how to handle this data. We assume that where projects have been mapped together by the Managing DP, they are co-financing arrangements.

We should show the list of the Managing DP’s projects where other organisations’ projects have been mapped to them. We should show:

* ~~total~~

each disbursement and commitment for the project by organisation (according to the AIMS data)

* ~~total~~

each disbursement and commitment for the project from the new IATI activities that have been assigned from other DPs

The user should be able to decide whether to accept or reject financial data from other DPs. ~~They should not be able to reject individual transactions, but they should be able to remove all disbursements and/or all commitments from individual DPs.~~ We should show the breakdown of transactions for now. If it’s too much data to show on the page (or if it isn’t clear) then we can show the total value of commitments or disbursements.

[see mockup “Co-financing”](http://test.brough.io/bd/sprint4.htm)

### 5. User management

As part of all of this, we will need to introduce the notion of a “user” to the interface. This will be the mechanism by which we will allow users to be responsible for their own projects and to pass responsibility over to other user.

If possible, it would be good to make use of the user management in the AIMS, but this should be a loose coupling so that the IATI import module remains somewhat generalised and could be applied in other systems that do not use exactly the same AIMS.

We should then create a simple welcome screen for a user which allows them to import their organisation’s projects - taking us to [the very first tab of sprint 2](http://test.brough.io/bd/sprint2.htm).

## Sprint 5

**STATUS: DRAFT**

* [Overview](http://bd-iati.github.io/development/sprint-5/#overview)
  + [Data integrity](http://bd-iati.github.io/development/sprint-5/#data-integrity)
  + [DP data overview](http://bd-iati.github.io/development/sprint-5/#dp-data-overview)
* [1. Import basic project data](http://bd-iati.github.io/development/sprint-5/#import-basic-project-data)
  + [Type of assistance](http://bd-iati.github.io/development/sprint-5/#type-of-assistance)
  + [Type of project](http://bd-iati.github.io/development/sprint-5/#type-of-project)
* [Activity dates](http://bd-iati.github.io/development/sprint-5/#activity-dates)
* [Implementation status](http://bd-iati.github.io/development/sprint-5/#implementation-status)
* [2. Import sectors](http://bd-iati.github.io/development/sprint-5/#import-sectors)
* [3. Import implementing organisations](http://bd-iati.github.io/development/sprint-5/#import-implementing-organisations)
* [4. Import project documents](http://bd-iati.github.io/development/sprint-5/#import-project-documents)
* [5. Import locations](http://bd-iati.github.io/development/sprint-5/#import-locations)
* [6. Handle updates](http://bd-iati.github.io/development/sprint-5/#handle-updates)

### Overview

The main goals of sprint 5 are:

1. Import non-financial data to the AIMS
2. Handle updates to other non-financial data

### Data integrity

* data in the live AIMS should not be touched
* data in the test AIMS can be changed and adjusted as necessary for the purposes of testing
* data in the IATI import tool can be deleted and re-generated at will throughout the development process, until we agree otherwise.

### DP data overview

For the purposes of this sprint, we will use data from a range of different DPs. The DPs are listed in [this Github issue](https://github.com/BD-IATI/edi/issues/46).

### 1. Import basic project data

In earlier sprints, we imported project titles and descriptions. We should now try to import the remainder of basic project data:

#### Type of assistance

This should be mapped from **Aid Type** as follows:

| **AidType code** | **Type of Assistance (AIMS)** |
| --- | --- |
| A01 | Budget Support |
| A02 | Sector Budget Support |
| B01 | Project Support |
| B02 | Project Support |
| B03 | Project Support |
| B04 | Project Support |
| C01 | Project Support |
| D01 | Project Support |
| D02 | Project Support |

#### Type of project

These should be mapped from **Aid Type** as follows:

| **AidType code** | **Type of Project (AIMS)** |
| --- | --- |
| A01 | Investment Project |
| A02 | Investment Project |
| B01 | Investment Project |
| B02 | Investment Project |
| B03 | Investment Project |
| B04 | Investment Project |
| C01 | Investment Project |
| D01 | Technical Assistance (TA) Project |
| D02 | Technical Assistance (TA) Project |

#### Activity dates

| **Activity Date Type (v1)** | **Activity Date Type (v2)** | **AIMS Date** |
| --- | --- | --- |
| start-actual or start-planned | 2 or 1 | Agreement Sign Date |
| start-planned | 1 | Planned Start Date |
| start-actual | 2 | Actual Start Date |
| end-planned | 3 | Planned Completion Date |
| end-actual | 4 | Revised Completion Date |

#### Implementation status

| **Activity Status code** | **Activity Status** | **Implementation Status (AIMS)** |
| --- | --- | --- |
| 1 | Pipeline/identification | Pipe Line |
| 2 | Implementation | On-going |
| 3 | Completion | Completed |
| 4 | Post-completion | Completed |
| 5 | Cancelled | Suspended |
| 6 | Suspended | Suspended |

### 2. Import sectors

If DAC sectors are used for this activity (where sector vocabulary is 1 or 2), map between them and BD gov sectoral classifications. Otherwise, enter no sectors.

See mapping file [CRS-BD-sector-codes.xls](http://bd-iati.github.io/development/CRS-BD-sector-codes.xls)

NB, this mapping file is incomplete and will be completed at a later date.

### 3. Import implementing organisations

Implementing organisations need to be mapped to the AIMS’ Executing Agencies. These use different tables depending on whether the Executing Agency is a Ministry, DP or NGO (I think?).

Organisation identifiers are not very good in IATI data, so we will need to ask the user to select the correct organisation. For DPs, we can try selecting them using the organisation ref which is stored in FundSource, though that will be imperfect. IATI does not have a standard methodology for identifying government agencies, while NGO identifiers are likely to be unreliable.

We should integrate this into existing step 3 in the following way:

1. ask if each implementing organisation is a DP, BD government, an NGO, or unknown (e.g. it may just state OTHER)
2. given the type of implementing organisation:
3. ask the user to select the organisation from a list (e.g. if they selected Govt, the list should show only government Ministries / Agencies).
4. if the implementing organisation is not present in the list, provide the user with the opportunity to add a new organisation (we will consider later adding an approvals process for new implementing organisations).
5. Also allow the user to state that the organisation is unclear (e.g. just states DONOR).
6. apply this to all projects
7. for those organisations where the implementing organisation is another DP, suggest that the project should be delegated to the other Managing DP, but allow the user to reject that.

We can try and help the user with some of these steps by:

1. guessing whether it’s a DP, BD government or NGO (using the organisation type attribute – [see codelist](http://iatistandard.org/202/codelists/OrganisationType/))
2. for those projects already mapped to the AIMS, we could try and use the existing extending organisations in the AIMS and link them together (though that will get complicated where there is more than one implementing organisation)
3. we could also try and match strings in the IATI data with the name in the AIMS.

We should get the basics running first and can then try to see if we can implement any of these steps to speed things up.

NB: there can be more than one implementing organisation in IATI data, and we should take each implementing organisation, not only one.

[See mockups](http://test.brough.io/bd/sprint5.htm) – the three steps on that page could be on different pages. Or the next step could only be revealed when having completed the previous step? E.g. “determine specific organisation” should show only if “determine organisation type” has been completed.

Please feel free to experiment with the user interface here if you think there is a better way of presenting this information.

### 4. Import project documents

Each project document, we should import, referring to the URL rather than using the file upload.

| **IATI Code** | **IATI Name** | **AIMS Name** |
| --- | --- | --- |
| A01 | Pre- and post-project impact appraisal | Project Document |
| A02 | Objectives / Purpose of activity | Project Document |
| A03 | Intended ultimate beneficiaries | Project Document |
| A04 | Conditions | Project Document |
| A05 | Budget | Project Document |
| A06 | Summary information about contract | Project Document |
| A07 | Review of project performance and evaluation | Annual Progress Report |
| A08 | Results, outcomes and outputs | Annual Progress Report |
| A09 | Memorandum of understanding | Financial Agreement |
| A10 | Tender | Others |
| A11 | Contract | Others |
| A12 | Activity web page | Others |

If classifications map to multiple AIMS names, map to OTHER (i.e. if you have a document with categories A01 and A02, map it to “Project document”; but if you have a document with categories A06 and A07, map it to “Other”).

### 5. Import locations

A couple of points on IATI locations:

* there is no percentage attribute in the IATI Standard, so the allocation should be calculated by the formula (100/number of locations)
* the specific administrative unit is rarely provided, so we probably have to map from locations.
* we could potentially use gazetteer references to obtain the correct location ID from [Geonames](http://geonames.org/), but not all DPs will use Geonames.

1. Check to see if we can determine the administrative level:
   * look at the administrative element (within each location),
   * if the vocabulary is G1, then check to see if the level is 1, 2, or 3 (mapping to Division, District and Upazilla respectively)
   * otherwise, set the location to be District.
2. Use [this function](http://www.plumislandmedia.net/mysql/haversine-mysql-nearest-loc/) (or something similar) to find the nearest location to the coordinates provided.´

### 6. Handle updates

Each evening when updating the data, if the data in IATI is not the same as that in the AIMS **and** we have chosen to take IATI data rather than AIMS data, then we should overwrite the AIMS data.

**LEAVE DOCUMENTS UNTIL WEEK 2**

We need to consider a bit more how to handle project documents where perhaps we would like to allow a mix of IATI and AIMS data.

## Sprint 6

**STATUS: CONFIRMED**

* [Overview](http://bd-iati.github.io/development/sprint-6/#overview)
  + [Data integrity](http://bd-iati.github.io/development/sprint-6/#data-integrity)
  + [DP data overview](http://bd-iati.github.io/development/sprint-6/#dp-data-overview)
* [1. Collect and display IATI financial data](http://bd-iati.github.io/development/sprint-6/#collect-and-display-iati-financial-data)
  + [Calculating planned disbursements: merging revised and original budgets](http://bd-iati.github.io/development/sprint-6/#calculating-planned-disbursements-merging-revised-and-original-budgets)
* [2. Inserting financial data to AIMS - first time](http://bd-iati.github.io/development/sprint-6/#inserting-financial-data-to-aims---first-time)
  + [Calculating exchange rates](http://bd-iati.github.io/development/sprint-6/#calculating-exchange-rates)
  + [Adding remarks](http://bd-iati.github.io/development/sprint-6/#adding-remarks)
* [3. Handling updates of IATI financial data](http://bd-iati.github.io/development/sprint-6/#handling-updates-of-iati-financial-data)
  + [Warning on mismatch between IATI and AIMS transactions](http://bd-iati.github.io/development/sprint-6/#warning-on-mismatch-between-iati-and-aims-transactions)
* [4. DP dashboard](http://bd-iati.github.io/development/sprint-6/#dp-dashboard)
* [Update: first week of sprint 6](http://bd-iati.github.io/development/sprint-6/#update-first-week-of-sprint-6)
* [Update: second week of sprint 6](http://bd-iati.github.io/development/sprint-6/#update-second-week-of-sprint-6)
  + [Importing transactional data for co-financed / trust fund projects](http://bd-iati.github.io/development/sprint-6/#importing-transactional-data-for-co-financed--trust-fund-projects)
  + [Handling updates for financial data for co-financed / trust fund projects](http://bd-iati.github.io/development/sprint-6/#handling-updates-for-financial-data-for-co-financed--trust-fund-projects)

### Overview

The main goals of sprint 6 are:

1. Collect and display IATI financial data
2. Insert financial data from IATI to the AIMS - the first time
3. Handle updates of IATI financial data, including tricky types of transactions, errors or unexpected changes in IATI data or the AIMS
4. Create an overview dashboard for the DP, including import logs and configuration options.

### Data integrity

* data in the live AIMS should not be touched
* data in the test AIMS can be changed and adjusted as necessary for the purposes of testing
* data in the IATI import tool can be deleted and re-generated at will throughout the development process, until we agree otherwise.

### DP data overview

For the purposes of this sprint, we will use data from Canada, DFID, Netherlands.

|  | **Canada** | **DFID** | **Netherlands** | **Asian Development Bank** |
| --- | --- | --- | --- | --- |
| Organisation identifier \* | CA-3 | GB-1 | XM-DAC-7 | 46004 |
| Version | 2.01 | 2.01 | 2.01 | 1.03 |
| Hierarchies | 1 | 2 | 1 | 1 |
| Languages | English, French | English | English | English |

\* this is currently the only- or most-used identifier. There may eventually be more than one identifier in use; see below.

The data from all three DPs is good. NB, Asian Development Bank data uses v1.03 of the Standard where the[location field is structured differently](http://bd-iati.github.io/documentation/versions/).

### 1. Collect and display IATI financial data

In sprint 3 (step 1) we built a “general field selection interface”, allowing users to decide whether data should be taken from IATI or the AIMS.

We need to extend that by breaking down “Financial data” field/section into three parts and providing a little more detail:

1. commitments
2. disbursements
3. planned disbursements.

For each category, we should show:

* the total value in that category (e.g. sum of all commitments)
* the amount per **Bangladesh fiscal year** in that category

For **commitments** and **disbursements**, in IATI, we should calculate this based on all transactions in each category. Remember that if we are looking at hierarchical projects, we should sum all transactions from all child projects, too. For **disbursements**, we should also include all **expenditures** in this category.

For **planned disbursements**, we should calculate this based on <budget> elements in all activities. We need some slightly more complicated logic here:

Within each activity:

* collect all original budgets
* collect all revised budgets
* for each revised budget:
  + if it overlaps **partially** or **wholly** with the original budget, take the revised budget and remove the original budget
  + if it does not overlap at all, then take the revised budget.

See the example below for an explanation of this logic.

#### Calculating planned disbursements: merging revised and original budgets

| **Budget type** | **Start date** | **End date** | **Value** | **Included?** |
| --- | --- | --- | --- | --- |
| Original | 2014-10-01 | 2014-12-31 | 50 | YES |
| Original | 2015-01-01 | 2015-03-31 | 100 | NO |
| Revised | 2015-01-01 | 2015-03-31 | 150 | YES |
| Original | 2016-01-01 | 2016-12-31 | 400 | NO |
| Revised | 2016-01-01 | 2016-06-30 | 300 | YES |
| Revised | 2016-07-01 | 2016-12-31 | 200 | YES |
| Revised | 2017-01-01 | 2017-06-30 | 150 | YES |

Having calculated this for each activity, in hierarchical activities, we would then take the amounts for each period in each activity and add them together.

* If there are budgets at hierarchy=1, then take only budgets from the hierarchy=1 activity. Ignore any budgets at hierarchy=2.
* Otherwise, and if there are budgets at hierarchy=2, then sum the amounts for each period in each activity.

### 2. Inserting financial data to AIMS - first time

If the user has chosen to take financial data from IATI, then we should update in the following way, for each category:

1. remove all financial data in the AIMS in that category
2. insert all new IATI transactions in that category

For example, if the user has chosen to import **commitments** then we should:

1. remove all commitments from the AIMS
2. insert each commitment in IATI as a new commitment in the AIMS

We can do this for a few basic fields to begin with. We should store:

* value (transaction/value/text())
* transaction date (transaction/transaction-date/@iso-date)
* transaction currency (see below)

#### Calculating exchange rates

We should use the Bangladesh Bank data to calculate exchange rates for each transaction.

We do not yet have daily rates available, so for now we should use the monthly average rates. However, we should prepare for having daily rates now.

* download data from Bangladesh Bank API for each month (by providing the last day of the month to the query)
* store DOLLAR\_PER\_CURRENCY rates for each currency and day. This could be in a flat file or a database, up to you.

Handling currency conversion for a particular transaction

* obtain the currency code, either from:
  + each transaction (value/@currency), OR
  + the activity (iati-activity/@default-currency)
* obtain the transaction value date (transaction/value/@value-date)
* convert the transaction value (transaction/value/text()) to USD by using the closest available currency conversion date to the transaction value date, for the relevant currency code
* store the currency conversion rate in the AIMS as well.

#### Adding remarks

In each transaction in the AIMS, we should add into the remarks section: Data automatically imported from IATI on YYYY-MM-DD

[see mockup, tab “Transactions”](http://test.brough.io/bd/sprint6.htm)

### 3. Handling updates of IATI financial data

On a nightly basis, we will want to automatically re-sync IATI financial data with the AIMS’ financial data.

The methodology for doing this is as follows:

1. Focus on projects where a link was established (in sprint 2) between the AIMS and IATI, and where the import preferences for that project (established in sprint 3) state that the financial data should come from IATI.
2. For each of the relevant categories of financial data (commitments, disbursement, planned disbursement) – again, which of these is relevant will depend on the import preferences for this project – collect two list of all transactions in IATI and AIMS respectively with:
   * transaction date
   * transaction value
3. If the AIMS contains any transactions not found in IATI, we should display a warning to the user that there was a mismatch (see below).
4. If IATI contains any transactions not found in AIMS – i.e. where the transaction date is not found in the AIMS – new transactions should be inserted.

#### Warning on mismatch between IATI and AIMS transactions

There could be several reasons why a transaction appears in AIMS and not in IATI:

1. The user has manually entered a transaction into the AIMS
2. Headquarters has removed a particular transaction from the IATI data (this should not happen, but it could)
3. Headquarters has updated all the transaction dates in the IATI data to be different dates, e.g. if they are providing cumulative transactions rather than actual ones (again, this should not happen, but it could)
4. Some error has occurred leading to all financial data disappearing from the IATI data (again, this should not happen, but it could theoretically happen either in the headquarters data, the IATI Datastore, or our import of the data).

When this happens, we should:

1. Pause automatic updates of the DP’s IATI data until the issue is resolved
2. Show an alert to the DP asking them to decide on a course of action.

The alert should show the difference between IATI and AIMS and the transactions that appear in AIMS but not in IATI.

The options for the user (for each project) should be:

1. Remove transactions from AIMS and import new transactions from IATI
2. Stop automatically importing IATI data and switch to taking data from the AIMS instead (this should update the project-specific import preferences)
3. Take no action now but remind me next time the data is checked (presumably the following night)

[see mockup, tab “Transaction merge conflict alert”](http://test.brough.io/bd/sprint6.htm)

### 4. DP dashboard

We are beginning to develop some quite complicated functionality at this point and it would be useful to develop a dashboard for each DP that shows what data has been imported from IATI, allows the DP to adjust import preferences, and shows any alerts.

We should show the following screens (either as tabs or as sections of the same page):

1. Latest data downloaded
2. List of imports / changes to particular projects
3. List of alerts for this DP (e.g. if something went wrong on import) – e.g. the above warnings if there is a mismatch in financial data
4. General import preferences and project-specific import preferences
5. List of activities in IATI that have not been imported to the AIMS (and option to begin import of those activities – which would take you to sprint 2)
6. List of activities from other DPs that have been delegated to you
7. List of your activities that you have delegated to other DPs (and ability to “recall” those activities if the other DP has not already mapped them to their own activitis in sprint 4)

Some of these screens can be placeholders for now.

[see mockup, tab “Donor landing page”](http://test.brough.io/bd/sprint6.htm)

### Update: first week of sprint 6

In the first week of sprint 6 (in reality, only three days), we will focus on the following areas:

* work with commitments and disbursements only (not planned disbursements / budgets for now)
* obtain and store exchange rates from the Bank of Bangladesh
* develop a simple screen that shows a summary of IATI transactions that would be imported to the AIMS, including the currency conversion rates and dates
* show a simple DP dashboard with mostly placeholder data.

### Update: second week of sprint 6

We have made great progress in the first part of this sprint and I think we have only one point remaining from above:

1. Updating and handling alerts on mismatch

In addition, we should add a couple of features to the DP Dashboard:

1. List parse errors for individual DPs’ data (preferably as detailed as possible - explaining what failed)
2. Import transactional data for co-financed and trust fund projects

Updating and handling alerts on mismatch is detailed above. It would be great to implement that, using the DP landing page to display the warnings.

Parse errors can similarly be displayed on the DP landing page, in line with the existing mock-ups.

[see mockup, tab “Donor landing page”](http://test.brough.io/bd/sprint6.htm)

Regarding importing transactional data for co-financed and trust-fund projects…

#### Importing transactional data for co-financed / trust fund projects

We already have a process for merging activities from different DPs – building on the [methodology outlined here](http://bd-iati.github.io/documentation/merging-updating-cofinanced-projects/).

Once we have merged financial data (in the current interface):

1. these preferences should be saved (i.e. do take DFID commitments, do not take Netherlands commitments)
2. the financial data that has been chosen to be included should then immediately be written to the AIMS. For co-financed projects, this should include individual transactions; for trust funds, it should update the total value of commitments held in the AIMS.

#### Handling updates for financial data for co-financed / trust fund projects

When the DP imports their own activity to the AIMS (or when the activity is updated), the import routine for financial data should respect the preferences set here.

For now, we can handle this by adjusting the financial data import [routine outlined above](http://bd-iati.github.io/development/sprint-6/#inserting-financial-data-to-aims---first-time) to be specific to an individual DP. For example:

* on updating / importing a UNDP activity, check the commitments / disbursements / etc. for that project in the AIMS which are tagged as from UNDP
* on updating / importing a DFID activity which has been mapped to a UNDP project in the AIMS (and where preferences have been set to import the DFID financial data), check the commitments / disbursements for the UNDP project in the AIMS which are tagged as from DFID.

## Sprint 7

**STATUS: DRAFT**

* [Overview](http://bd-iati.github.io/development/sprint-7/#overview)
  + [Data integrity](http://bd-iati.github.io/development/sprint-7/#data-integrity)
  + [DP data overview](http://bd-iati.github.io/development/sprint-7/#dp-data-overview)
* [1. Allow components of projects to be deselected](http://bd-iati.github.io/development/sprint-7/#allow-components-of-projects-to-be-deselected)
* [2. Improve manual mapping interface](http://bd-iati.github.io/development/sprint-7/#improve-manual-mapping-interface)
* [3. Allow projects to be grouped](http://bd-iati.github.io/development/sprint-7/#allow-projects-to-be-grouped)
* [4. Save progress as the user works through the stages](http://bd-iati.github.io/development/sprint-7/#save-progress-as-the-user-works-through-the-stages)
* [5. Speed up performance](http://bd-iati.github.io/development/sprint-7/#speed-up-performance)
* [Remaining to do](http://bd-iati.github.io/development/sprint-7/#remaining-to-do)

### Overview

The main goals of sprint 7 are:

1. Allow components of projects to be deselected (required for DFID projects)
2. Allow projects to be manually mapped to the AIMS where automatic mapping failed
3. Allow activities to be grouped into recognisable projects (required for USAID projects)
4. Save progress as the user works through the stages
5. Speed up performance

### Data integrity

* data in the live AIMS should not be touched
* data in the test AIMS can be changed and adjusted as necessary for the purposes of testing
* data in the IATI import tool can be deleted and re-generated at will.

### DP data overview

We’re now using data from as many DPs as possible. In this stage, we will improve the way we are handling DFID and USAID data in particular.

### 1. Allow components of projects to be deselected

In sprint 2 we allowed users to tell us which hierarchy of projects they should map from. However, where there are multiple sub-components of a project, the user needs to be able to deselect some of those sub-components.

After step 2 (Filter Bangladesh-relevant activities) we should therefore add in an additional step if:

1. a donor has a hierarchy in their activities
2. the donor has not selected the lowest level

We should then show each of the projects and their subcomponents, with a checkbox to allow the subcomponents to be deselected.

[See mockup (“2.1. Select project components”)](http://test.brough.io/bd/sprint7.htm)

### 2. Improve manual mapping interface

At the moment, step 5 is quite confusing. We should merge it with step 4 in the following way:

1. In the IATI activities not found in the AIMS section, there should be a drop down box of all projects from that DP in the AIMS, with the top option being New project.
2. Selecting an AIMS project to map to should also remove the relevant projects from the IATI activities not found in the AIMS section and display it in the IATI activities matched to the AIMS section.
3. Clicking the unmatch button next to a project in the IATI activities matched to the AIMS section should return that project to the IATI activities not found in the AIMS section.

[See mockup (“4. Select project components”)](http://test.brough.io/bd/sprint7.htm)

### 3. Allow projects to be grouped

In sprint 2 we developed an interface to allow projects to be grouped. We need this in particular for USAID where there are a large number of activities that should be grouped to make recognisable projects. However, this interface is currently very complex and would be slow for a large number of activities, so we will remove this interface and simplify the functionality.

We can achieve this in step 4, under IATI activities not found in the AIMS.

* We should allow multiple IATI projects to be mapped to a single AIMS project.
* We should allow the list of projects to be filtered by extending organisation (using a drop-down or typeahead interface)
* We should also allow the list of projects to be filtered by project ID (try typing in AID-388 to the “Project ID” filter, for example)

Step 5 (IATI activities matched to the AIMS) can now be removed as we have now incorporated the functionality elsewhere.

[See mockup (“4. Select project components”)](http://test.brough.io/bd/sprint7.htm)

### 4. Save progress as the user works through the stages

It should be possible for a user to get to each of the stages and return (e.g. the following day, or after closing their browser) to the point they were last at. This will require saving the state of their progress as they move through the interface. On the Dashboard, there should also be a button to return to their last-saved progress.

As a side-effect of this, there could be a set of projects that are in the middle of being processed when a new project appears (e.g. the new project is created overnight). Projects in the middle of being processed should be completed before beginning to process new projects.

### 5. Speed up performance

At the moment the import module is quite slow as we are sending a lot of data to the browser. This is deliberate as it has made development more straightforward, but it needs to be sped up significantly before we move to use the module in production.

### Remaining to do

Once we’ve completed the other steps identified above, we need to go through and check the interface works for a range of different donors, and ensure the user interface is clear and clean. This will probably involve some textual and stylistic modification. We also need to check that updates are working well and that the project delegation / merging functions work well.

# Documentation

This section contains documentation on the technical specifics and process recommendations for sustainable IATI import.

The technical specifics will evolve over the course of software development. This section will be used as a repository of our latest thinking. We welcome feedback from others – in particular, those that have worked to import IATI data to country systems previously.

## Retrieving IATI data

### Introduction

When loading data into the IATI-AIMS import tool, users should not be required to look at or handle XML files. This must be handled behind the scenes.

We’re proposing to retrieve the data from the IATI Datastore in the first instance. The Datastore makes it possible to query for subsets of IATI data according to a number of filters.

### Retrieving from the IATI Datastore

To retrieve data from Bangladesh for a particular donor, we can run the following query:

http://datastore.iatistandard.org/api/1/access/activity.xml?recipient-country=BD&reporting-org=CA-3&stream=True

In this instance, we’ve chosen to receive all activities tagged as Bangladesh from Canada’s DFATD. The parameters should be adjusted as follows:

* reporting-org: the organisation identifier for the reporting organisation you’re interested in. This maps to reporting-org/@ref in IATI data. In the example above, we’ve chosen CA-3 for Canada DFATD.

### Troubleshooting

The Datastore has some [known issues](https://github.com/iati/iati-datastore/issues).

If it’s not possible to use the IATI Datastore, we can also try using [OIPA](http://www.oipa.nl/api/v3/docs/).

## Hierarchies of projects

The data [retrieved from the datastore](http://bd-iati.github.io/documentation/retrieving-data/) will be almost identical to the way individual donors publish their own data. This poses something of a challenge, as donors can publish in different ways, sometimes using hierarchies of activities if that is important to accurately reflect their business model.

### Understanding hierarchies in IATI

Some donors (currently a handful) choose to structure their data according to a hierarchy of activities. For example, DFID has the following two projects and several sub-components:

* GB-1-107368 **Rural Electrification Development Project**
  + GB-1-107368-101 PROCOFSERVICES and P0050 for Rural Electrification Development Project
  + GB-1-107368-102 MULTILATORGANISATION and P0050 for Rural Electrification Development Project
  + GB-1-107368-103 NONBUDSUPPFINAID and P0050 for Rural Electrification Development Project
  + GB-1-107368-104 MOU With USAID for REDP
  + GB-1-107368-105 NONBUDSUPPFINAID and CP026 for Rural Electrification Development Project
* GB-1-107369 **Health, Nutrition and Population Sector Programme (HNPSP)**
  + GB-1-107369-101 NONBUDSUPPFINAID and P0050 for Health, Nutrition and Population Sector Programme (HNPSP)
  + GB-1-107369-102 PROCOFSERVICES and P0050 for Health, Nutrition and Population Sector Programme (HNPSP)
  + GB-1-107369-103 MULTILATORGANISATION and P0050 for Health, Nutrition and Population Sector Programme (HNPSP)
  + GB-1-107369-104 Procurement of Goods for Health, Nutrition and Population Sector Programme (HNPSP
  + GB-1-107369-105 Programme Capital for Health, Nutrition and Population Sector Programme (HNPSP)

The activities highlighted in **bold** are DFID projects (hierarchy=1) that contain five components (hierarchy=2) each. All of these units of aid – projects and components – are held within separate iati-activity. Each unit of aid is then related to its parent or child activities by the related-activity tag.

## Relating this to the AIMS

The above relationship between projects and components is both important and useful. In DFID’s data, hierarchy=1 activities contain project documentation whereas hierarchy=2 activities contain financial transactions. Hierarchy=1 activities have more meaningful titles to most of the world – but the sub-components are helpful for the donor to see on import because they group spending together and may have meaning to them.

In Bangladesh, projects are only reportable to the AIMS at a certain point, after they have been approved by the government, but some donors will begin preparatory work in advance of this approval. Currently, when DFID report to the AIMS (manually), they select the sub-components that are reportable and exclude those that are not.

We can do the same here.

## How to apply this to the AIMS

We should:

1. Determine that a hierarchy of projects has been provided
2. Allow the user to select which level they would prefer to report at (in the case of DFID, whether they would prefer to publish projects or components) – providing some sample data to help decide.
   * We would preferably also show the % of activities at each hierarchy that have been found in the AIMS and provide a recommendation - e.g.

*we found 90% of hierarchy=1 activities in the AIMS and 0% of hierarchy=2 activities in the AIMS. We recommend you group by hierarchy=1 activities.*

1. Allow the user to exclude the projects or sub-components that should not be included.

## Importing this data

Once we have related a series of activities (projects and sub-components) to a particular AIMS project, we can proceed in the normal way by allowing the user to choose which data should be imported.

## Project types

Projects are the standard unit of aid in the AIMS with most attributes being assigned at the project level.

### Financing relationships in the Bangladesh AIMS

The Bangladesh AIMS allows for three financing relationships for projects:

| **Type of project** | **Description** |
| --- | --- |
| (1) Standard projects | These have one funder, and one or many implementers (in many cases, the funder is also the implementer). For these projects it is clear who is responsible for entering data, and to whom the project (and associated financial flows) are attributed. |
| (2) Co-financed / pooled projects | These have several funders and one or many implementers. It is know how much each funder is contributing to the project i.e. the funds are earmarked. The project data may be available in IATI from several funders and implementers. The Bangladesh AIMS requires all collaborating funders to select a managing donor between them. This donor will take responsibility for managing the project in the AIMS.   The IATI project (and financing information) from all reporters can be assigned to the same AIMS project for the managing donor to manage. Users will need to be able to select the managing donor from a list if the project does not already exist in the AIMS. |
| (3) Trust Fund financed projects | Several funders may also collaborate to fund multiple projects. When the funds they supply are mixed in a joint pot (from which the different projects are funded), it is not possible to ascertain exactly which funder has contributed to each project financed from the pot. The funds are unearmarked. In the Bangladesh AIMS, this situation is dealt with by setting up a Trust Fund, available [here for users with sufficient privileges](http://aims.erd.gov.bd/AIMS/TrustFund/Index). A Trust Fund is like a specific donor for that pot of money and each trust fund is allocated a managing donor.  Users will need to be able to select (or add) a Trust Fund as the responsible entity for the project which will give management rights for the project to the managing donor fo the selected Trust Fund. |

The main challenge will be with types (2) and (3), where projects are likely to be reported at multiple stages by different organisations. Given that we are not at the outset proposing to focus on NGO data, we are less likely to face any problems with type (1).

### Distingushing between types of projects in IATI data

The IATI Standard allows for traceability of a project, allowing publishers to specify that they received money from another organisation, or gave money to another organisation. However, in reality, these fields as still not filled out by many organisations so we need to find other ways of identifying these projects and reconciling projects published at multiple points.

We therefore propose to focus on two parts of the IATI Standard:

* **Aid type** – which states several broad categories of the type of project.
* **Organisations** – if it looks like an implementing organisation is also reporting elsewhere in the AIMS.

#### Aid type

The [Aid Type](http://iatistandard.org/201/codelists/AidType/) codelist provides a series of codes for different types of projects. This code is generally found indefault-aid-type/@code on each activity.

* Standard projects (type (1) above) should have their aid type set to C01.
* Cofinanced / pooled projects projects may often have aid type set to B04.
* Trust Fund financed projects may often have aid type set to B03.

We therefore propose to flag those projects with a code other than C01 for review by users, assuming that they are probably more complicated projects that need to be handled with care.

#### Implementing organisation

Implementing organisations are harder to handle because there is no solid methodology for consistently identifying organisations. There are a couple of codes in the Organisation Type codelist that may help, but we can’t rely on them. Our main goal here is to identify if an organisation appears to be found elsewhere in the AIMS. There are a couple of ways to handle this:

1. projects should be presumed to be found elsewhere in the AIMS if:
   * the implementing organisation is of organisation type 40 (“multilateral”);
   * the implementing organisation has a ref 11000 (“donor governments”) or 13000 (“third-country governments, delegated cooperation”)
   * the implementing organisation has a ref that matches a “fund source” from the AIMS – NB there is a field to state the IATI identifier of the organisation, but it needs to be filled out
2. users should be presented with the list of implementing organisations contained in all of the projects (a single unique list) and asked to correct the answers from step 1.

We may also want to consider looking at provider-org and receiver-org on transactions, but the data is likely to be of a much lower quality than that found in implementing organisations.

## Financial data

The Bangladesh AIMS contains a number of different sorts of financial data. IATI also has a number of different sorts of financial data, and has a couple of different ways of handling such data. Some donors may interpret this a little differently. Given these differences, the import tool will need to provide users with some information (e.g. comparing amounts in two systems) in order to help them make sensible decisions.

### Financial data in the AIMS

Financial data is found in the AIMS either at the project level (as a single value) or as a series of transactions of different types, under funding information.

#### Project information

A single value (in different currencies, with a [currency conversion](http://bd-iati.github.io/documentation/currency/) rate to USD) can be provided for each of the following fields:

* Project Cost in USD (total project cost)
* DP Contribution (managing DP’s contribution)
* Other Contribution (other DPs’ contributions)
* GoB Sharing (government contribution)

#### Funding information

Many values (in different currencies, with currency conversion rates to USD) can be provided for each of the following fields:

* Commitments
* Planned disbursements
* Actual Disbursements
* Expenditure

Disbursements (both planned and actual) are linked to commitments (which are actually the value of financing agreements)

### Financial data in IATI

Financial data can be found in two places in IATI.

Most financial data is found in <transaction> elements. Each activity can have many transactions. Transactions can be of several different types. The main three we will be concerned with are commitment, disbursement andexpenditure. There are also a series of transaction types relating to loans; we will look at these in more detail when the need arises.

Financial data relating to forward plans is found in two other elements, <budget> and <planned-disbursement>. The difference between these two is not well documented and usage varies. **Budget** is generally supposed to be medium-term forward plans for an activity – at least one year ahead, and broken down into quarterly chunks.**Planned disbursement** is generally supposed to show the specific date when specific tranches of funding will be disbursed. It is likely to be much more short term and particularly used in the context of budget support.

### Mapping financial data from IATI to AIMS

#### Mapping to funding information

Planned disbursements are handled somewhat differently –[*see below*](http://bd-iati.github.io/documentation/financial-data/#mapping-to-planned-disbursements).

| **AIMS** | **IATI v1.x** | **IATI v2.x** | **IATI definition** |
| --- | --- | --- | --- |
| Commitment | C | 2 | A firm, written obligation from a donor or provider to provide a specified amount of funds, under particular terms and conditions, for specific purposes, for the benefit of the recipient. |
| Actual Disbursement | D; E | 3; 4 | D: Outgoing funds that are placed at the disposal of a recipient government or organisation, or funds transferred between two separately reported activities. E: Outgoing funds that are spent on goods and services for the activity. |
| Actual Expenditure | – | – | see notes |

Notes:

* it appears that in both the AIMS and IATI, donors use Disbursement and Expenditure interchangeably and inconsistently. We recommend in general mapping both of these types of transactions to Disbursements. In IATI, expenditures are sometimes used for administrative or ancilliary costs, or for funds spent on things that would not be expected to show up lower down the chain of IATI publishers – for example, office supplies. However, the definition and distinction is not very clear.
* there is not total agreement in IATI about the definition of “Commitment”. While the original DAC definition (which the IATI definition is derived from, and fairly dependent upon) defines it as a “firm, written obligation” – normally a formal letter exchanged with the government – it is sometimes also understood to be the total value of the project. The budget element is probably more appropriate for the total value of the proejct, but we need to be mindful that donors may use this data in slightly different ways.

The two IATI columns show the codes used on the transaction-type/@code attribute on each transaction.

#### Mapping to planned disbursements

We will generally map from budget to planned disbursements. We need to have a definition of what is required in the AIMS (e.g. must planned disbursements be disaggregated by quarter). We should consider how forward data is currently used in the AIMS, and what happens to “forward” data once the date is past.

#### Mapping to project information

| **AIMS** | **IATI** |
| --- | --- |
| Project Cost in USD | Sum of all budget elements (though see note above - we may need to consider commitments for this sometimes) |
| DP contribution | Sum of all budget elements for the managing DP |
| Other contribution | Sum of all budget elements for other contributing DPs |
| GoB sharing | No mechanism for capturing this data in IATI – must be manually filled out in the AIMS |

## Updating data

### Updating activities

Where a field has been tied to a specific IATI activity, it should be possible to update that field automatically as IATI data changes. IATI data should be checked for updates each night. To begin with, a user should be prompted when a relevant change has been identified. They should then be able to choose whether to accept the changes or reject them. Their choices should be persistent if the update is rejected, i.e. the following night, even though the IATI data differs from the AIMS data, the user should not be prompted to update. Notifications could be provided periodically be email or by using the AIMS’ notifications interface.

We assume that users will generally want to update fields that have been sourced from IATI data, but not where fields have been sourced from AIMS data. However, at least to begin with, users should always have to choose to import data rather than for it to occur automatically without prompting. In time, as more analytics are collected on the way the import tool is being used, users could decide to allow the data to automatically flow in.

The following steps could be used to check for and manage updates:

1. on a nightly basis, download relevant data from the IATI Registry;
2. record activities that are not matched at all in the AIMS. Flag those originating from DPs that have imported some data from IATI to the AIMS (they could be new activities);
3. where an IATI activity is linked to an AIMS activity, check to see if any of the matched fields have different values. This could be achieved by comparing values in the AIMS with the values in the most recent IATI data, or it could be achieved by comparing the most recent IATI data with the previously downloaded IATI data.

Different approaches should be taken depending on the nature of the change:

1. if there are new activities, present the user with the option to begin importing those activities (following the methodology outlined above – beginning at stage 1);
2. if there are differences in matched fields, present the user with the option to update those fields with the new values;
3. if there are new fields that were previously unknown to the AIMS (e.g. there is now data on the location of activities), present the user with the option of importing that data to their activities (following the methodology outlined above – beginning at stage 2);
4. if activities are deleted, alert the user. Provide the option to remove the activities from the AIMS or unlink them from the IATI data. It is also important to state how many other activities were deleted – it could be a technical error that has caused the activities to be deleted. We should be particularly careful about deleting information and would suggest not doing so automatically for the foreseeable future.

It will probably be desirable to develop a distinct module that can compare and record differences in IATI data files, as well as expose that data in an intuitive way.

### Updating transactions

Financial transactions should be handled in a similar way to activities – using the same methodology for comparing and recording differences in IATI data files. However, given the potential large volume of changes to financial transactions, users will need to be presented with an aggregate overview of the nature of the changes, so that they can make an informed decision about the implications of any changes to the transactions, and whether they would like to import them.

Financial transactions should generally not be overwritten or deleted from the AIMS; IATI data should be assumed “append-only” in this respect. However, it is important to be aware of and handle potential exceptions to that rule (e.g. where an organisation changes the dates of transactions after the fact). We will consider the best approach to handling exceptions to this – whether it is better to try to adjust the data in the user interface or to work with donors to fix their data.

## Versions in IATI data

### Versions of the IATI Standard

There are now a number of versions of the IATI Standard. The main breaking changes are:

* Versions 1.03-1.04 – substantial change to the sub-national location / geocoding schema ([see changelog](http://iatistandard.org/upgrades/decimal-upgrade-to-1-04/1-04-changes/)). One of the main changes was where location coordinates were stored. 1.03 placed longitude and latitude as separate attributes on acoordinates element:
* <coordinates latitude="31.616944" longitude="65.716944" precision="2" />

Whereas 1.04 places the coordinates together in the text of a point/pos element:

<point srsName="http://www.opengis.net/def/crs/EPSG/0/4326">

<pos>31.616944 65.716944</pos>

</point>

See [location in 1.03](http://iatistandard.org/103/activities-standard/location/index.html) and [location in 1.04](http://iatistandard.org/104/activity-standard/iati-activities/iati-activity/location/)

* Versions 1.x to 2.x – two major changes:
  + a number of English-language codelists were changed to numeric codes. The most important of these are:
    - [activity date type](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#activity-date-type-amended-codes)
    - [organisation role](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#organisation-role-amended-codes)
    - [sector vocabulary](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#sector-vocabulary-was-vocabulary-amended-codes)
    - [transaction type](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#transaction-type-amended-codes)
  + changes to improve the way the Standard handles publication in multiple languages.
    - The main change here was to include a narrative element under [various elements](http://iatistandard.org/201/upgrades/integer-upgrade-to-2-01/2-01-changes/#narrative-new-elements), which then contains the text that would previously have been just within the text of the elements themselves.
    - For example, in v1.x, Canada’s titles looked something like:
    - <title xml:lang="en">Legal Reform</title>
    - <title xml:lang="fr">Réforme juridique</title>

Whereas in v2.01, the data looks something like this:

<title>

<narrative xml:lang="en">Legal Reform</narrative>

<narrative xml:lang="fr">Réforme juridique</narrative>

</title>

### Approach to handling different versions

IATI data retrieved from different donors may be available in different versions of the IATI Standard. A component to automatically convert all data to a single version of the IATI Standard will make it significantly easier to handle the data later in the application. We will take version 2.02 of the IATI Standard as our basis, as it’s the most recent version of the Standard. IATI-XML data from the AIMS should also be passed through this component. The import process of matching data from different sources will become significantly easier once all the data is in the same format.

This component should:

* Allow data from different IATI-XML versions to be imported and converted to a standard JSON representation, including nesting hierarchical activities
* Be available as a distinct module so that it can be used and maintained by other users of IATI data.

### Standardising and converting data

There are two different ways we could do this:

1. Standardise XML, then convert to JSON:
   * Create a converter from each version of the IATI Standard to version 2.02
   * Create one converter from version 2.02 of the IATI Standard to a standard JSON representation
2. Convert to standard JSON:
   * Create a converter from each version of the IATI Standard to a standard JSON representation of the IATI data.

We will proceed with option 1 (standardise, then convert) as it is likely to involve less duplication in the code, and also because it is likely to be useful to other users who will need to handle data from different versions of the IATI Standard.

## Modifications to the AIMS

### Reading from the AIMS

In order to reconcile projects from IATI with existing projects, it will be necessary to read data out of the existing database and provide access to it via an API.

The API will need to provide the following functionality:

1. list projects according to a series of set queries, including:
   * by DP
   * by sector
   * all data (all DPs and all sectors)
2. list trust funds
3. list DPs (name, identification)

It may be necessary to add additional filters in time, so a flexible approach will be required for querying the data.

Where project data is provided, the data should be provided in the IATI-XML format so that it can be read in by the front end in the same way as IATI data published by donors. There is an additional benefit in doing this in providing IATI export functionality from the AIMS.

In order to do this, the software supplier will require access to the AIMS source code.

### Writing to the AIMS

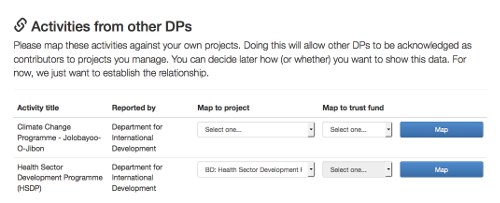
In order to implement the decisions taken in the IATI import tool, it will also be necessary to have the ability to write into the AIMS. This will be a question of updating the data in the AIMS for a particular project, by mapping project data from the IATI import tool to specific fields. The AIMS’ **notifications interface** may be useful in integrating the IATI-AIMS import tool into the AIMS.

It may be necessary to make some adjustments to the database structure in order to record the provenance of data and allow for it to be automatically updated in the future. The nature of these adjustments will depend on the way the existing database is structured, as well as the effects of other business logic in the source code.

In order to do this, the software supplier will require access to the AIMS source code under the same conditions as stated above.

## Merging and updating cofinanced projects

### Collecting and storing mappings

*Mapping DFID activities against World Bank projects and Trust Funds*

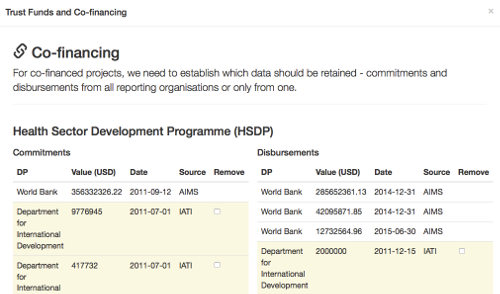
We currently store information about the mapping between projects and “delegated” activities. Other DPs can delegate an activity to the Managing DP which can then merge those activities into its own project. This is important where, for example, Canada and DFID publish activiites which are there contributions to a World Bank project. We need to avoid double-counting, but we may also want to take some of this data in order to ensure that each DP’s contributions are accurately reflected.

We currently store this data by including the other activity on the (delegated) activity. For example: DFID contribution GB-1-12345 of £500 to a World Bank-managed co-financed project 44000-P56789 would be recorded as:

**Activity** Table: project GB-1-12345

* MappedCofinancedActivity: 44000-P56789

### Collecting and storing preferences

*Merging DFID and World Bank financial data for a co-financed project, the Health Sector Development Programme*

We need to store preferences about which data should be collected. In the above image, we have shown all transactions from a delegated project (from other DPs) and the main project (from the managing DP).

We need to instead show:

* the total amount (for each of commitments, disbursements, planned disbursements) for each activity mapped from another DP
* a checkbox to allow that data to be included or excluded.

We then need to write this data to the module, to record something like the following table:

**DelegatedActivityPreference** Table

| **Activity ID** | **Field Name** | **Include** |
| --- | --- | --- |
| GB-1-12345 (Foreign Key) | Commitments | True |

NB: we should do this both for co-financed and Trust Fund projects. However, for Trust Fund projects, only preferences for **commitment** values should be retained, as this is the only data accepted by the AIMS.

### Displaying preferences

We can then use these two tables to show, for each co-financed project:

1. Delegated activities related to this co-financed project
2. Preferences about whether commitments, disbursements, and planned disbursemetns should be included for that co-financed project.

We should do the same for Trust Funds; however, in Trust Funds, we will only show Commitments as that is the only data that is accepted in the AIMS.

On the Dashboard page, we can then (probably in the bottom-right) show a list of activities for this DP which have at least one other activity mapped/delegated to them. That list should provide a pop-up displaying the above preferences, and allow them to be edited.

## Currency conversion

The primary currency used in the AIMS is USD, though it also has mechanisms for conversion to BDT and a host of donor currencies. Given that the AIMS already has a way of handling currency conversion, this does not all need to be performed in the IATI import tool front end. However, it is important to understand how it works and how the IATI import tool can help to make life easier.

### Currency conversion rates to BDT

In accordance with the AIMS, we will generally use the [Bangladesh Bank](https://www.bb.org.bd/) (the central bank) rates. These could be programmatically retrieved from the following site, though an API returning JSON would probably be preferable:

https://www.bb.org.bd/econdata/exchangerate\_dtl.php?loadmode=2&cboCurrency=All&ddlYear=2016&UsersList=January&SelectPeriod=January,%202015

The value/@value-date attribute in IATI data provides the specific value date for currency conversion. We should find the exchange rate at the closest date in the Bangladesh central bank data to the value/@value-date

The Bangladesh Bank [explains how these rates are calculated](https://www.bb.org.bd/econdata/exchangerate.php):

*Inter-bank exchange rates are also used by BB for purchase and sale transactions with the Government and different International Organizations. The USD/BDT buying and selling rates below are highest and lowest inter-bank exchange rates at Dhaka. The cross rates of BDT with other foreign currencies are based on NY and Dhaka closing exchange rates.*

 In some cases, it is important to know the exact exchange rate according to the donor - e.g. for debt sustainability analysis. This information is not currently available in the IATI Standard, and will have to be collected manually.

### Currency availability in central bank data

The following currencies are available in central bank data:

* USD
* EUR
* GBP
* AUD
* JPY
* CAD
* SEK
* SGD
* CNH
* INR

### Other currencies

 We need to have a source for other currencies. What should this source be?

## Glossary

| **Acronym** | **Definition** |
| --- | --- |
| DP | Development Partner |
| IATI | International Aid Transparency Initiative |
| UoA | Unit of Aid |
| AIMS | Aid Information Management System |
| OECD | Organisation for Economic Dooperation and Development |