

May 2020

Beyond Certification: Building Productive Relationships with Source System Vendors

Rosh Dhanawade, Data Architect
Molly Stewart, Research Scientist
[INsite Consortium at Indiana University](#)¹

The target audience for this white paper is a local education agency (or similar type of organization) with a functioning Operational Data Store (ODS). All technical terminology used in this paper should be familiar to someone who has reached the point of having a functioning ODS and Ed-Fi API. One technical assumption we make in this paper is that your ODS and API are core, not extended—or, if there are extensions, that you are fully aware of what those are. We do not assume expert-level knowledge, but enough to have meaningful conversations.

Vendor relationships with education LEAs (or similar organizations) can fall on a spectrum anywhere from harmony to hostility. However, the tenor of these relationships is not random; as an educational organization, there are several ways in which you can help these relationships be efficient and effective in terms of meeting your users' needs regarding data populating the ODS. It is also important to note up front that vendors do have the right to refuse Ed-Fi-related requests and/or request payment for them. These requests often require developer time and workload and may not make sense within the vendor's internal development timeline. However, there are ways to make it more likely that a vendor will grant your request(s) without additional charges. In this paper, we outline four overarching strategies for building fruitful relationships with vendors:

Do your research

Find the right person

Make the value proposition

Meet them halfway

First and most importantly, do your research, specifically regarding your ODS structure. This strategy is also applicable to a wide variety of Ed-Fi implementation aspects and will never be a

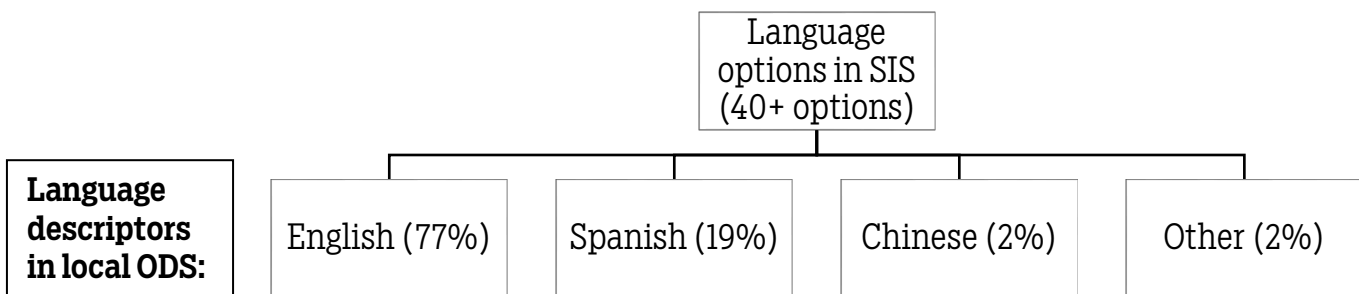
¹ The authors thank our grant funders, the Michael and Susan Dell Foundation, as well as Megan VanDeventer and Caroline Kazmierski, who provided feedback on this paper. All opinions and errors are the authors' own.

waste of time. Second, make sure that you have the right person on the other end of the conversation. Third, make it clear to the vendor that there is a benefit to them of meeting your particular request. This strategy should be an organic outgrowth of doing sufficient research, but it also requires additional work and framing on your part. Finally, specifically regarding assessment vendors without an existing native integration, we discuss strategies that you can employ to significantly lessen the burden on the vendor of building out an API (and therefore make it more likely that the vendor will meet the request).

Do your research

The first and most fundamental strategy for successful vendor relationships is to do your own background research on the structure of your ODS and the specific use cases that your organization is focused on. If you request data to be sent from the vendor, and that kind of data does not have a landing place in the API, then you (1) look silly, (2) may be telling your users wrong information about how the integration works and what data can be available, and/or (3) risk eroding trust with vendors because you have not done your part to make sure that your request is possible. You may assume that vendors have the API specification memorized, but they likely do not, and if they waste their time trying to send data that has no landing place, they probably will not be interested in entertaining your future requests. The rest of this section provides detail for the beginning Ed-Fi API user in terms of how to get a better understanding of the structure of your ODS and the data that can be transmitted via the Ed-Fi API.

Connecting a vendor system with an ODS requires some reconciliation of the way you manage and record data locally with the way that the ODS is going to store the information. In order to understand how to make those decisions, you will need to understand how the ODS receives and stores information from the Ed-Fi API. At minimum, you need to understand descriptors and types as well as the locations within the ODS that hold the data you care about. In Version 2.x, which is what we use, you can create descriptors that can match district codes one-to-one (or close to one-to-one). Types allow you to map those descriptors (which may be many, depending on the district's local coding) to codes that are more general and useful for certain kinds of data aggregation. For example, K-12 students in Indiana primarily speak English and/or Spanish; however, the remaining 4.2% who speak neither English nor Spanish speak one or more of 37 other languages. The following diagram provides one way in which a local district might design their language descriptors:



School districts need to record and understand how many of their students speak specific languages, but in terms of aggregating patterns, many data needs will be met by a smaller number of types. The type/descriptor structure allows us to combine certain languages into additional types in the future, such as a “Nordic” language type. The decision to create these types is context-specific, based on the demographics of a local Ed-Fi implementation. It is also key to understand that types and descriptors can appear in multiple tables in the ODS, so you need to understand how those tables relate to each other, what data each table has, and why the table is using a descriptor versus a type.²

When doing your research, also keep in mind that you may not be on the latest version, and that your version may have different options than the latest documentation. For example, version 3.x no longer uses types in the same way that 2.x did, so you can forget everything we just said about types!³ Fortunately, there are version-specific resources, including the Ed-Fi Unifying Data Model (“data standard”), the Ed-Fi Data (“MetaEd”) Handbook on the Ed-Fi tech docs website, and the API documentation on your own Swagger site. Every implementation of Ed-Fi will have its own Swagger site with API documentation that is specific to your implementation, including any extensions that you may have created.⁴ The Swagger site has the names of the API resources⁵ as well as their definitions, which give a general idea of the kind of data that can be sent there (consider the analogy of a shape-matching toy—ODS fields can only accept data that is in a specific format). It is designed to be straightforward and understandable even by less tech-savvy users; an educator should be able to understand those terminologies and definitions, at least generally (e.g., “accountabilityRatings,” “feederSchoolAssociations”), which can be helpful for conversations between your integration team and the educators whose data needs you are trying to meet. Your Swagger site is version-specific and should be a reference that you continually revisit in order to make sure that your ODS is capable of receiving certain data points from the API. Use the Meta-Ed handbook in conjunction with this; the handbook is basically a dictionary that describes in more detail the terms that you see in Swagger.

You should also review your own ODS database directly to see how the data looks when it lands; generally, you will want to be checking that the format expected by the ODS/API matches how the source system is sending the data. Specifically, you will want to notice whether there are unexpectedly empty fields (see the figure below for an example) and/or whether fields are



² For more in-depth information on how to navigate the ODS structure, check out the INsite Operational Data Store webinar at <https://education.indiana.edu/community/insite/resources.html>

³ Ed-Fi TechDocs has a great resource that compares the three technical versions at a glance: <https://techdocs.ed-fi.org/display/ETKB/Ed-Fi+Technical+Suite+Version+Matrix>

⁴ For example, here is the INsite swagger site: <https://api-st.insite.indiana.edu/ST-Swagger/index.html?url=https://api-st.insite.indiana.edu:443/ST-API/metadata/resources/api-docs#/>

⁵ For those readers who may not be tech-savvy: “A resource is an object with a type, associated data, relationships to other resources, and a set of methods that operate on it” (<https://restful-api-design.readthedocs.io/en/latest/resources.html>)

populating with unexpected data types. Data fields where you will commonly see errors are dates and phone numbers. The ODS will require these to be in a particular format, but a source system might allow a user to manually enter data (as opposed to having a dropdown calendar or 3-3-4 limited entry fields for phone numbers). Hand-entered data that do not match the expected format in the ODS/API will result in errors.

The screenshot shows a SQL Server query window with the following SQL code:

```

SELECT [StudentUSI]
,[SchoolId]
,[SchoolYear]
,[EntryDate]
,[EntryGradeLevelDescriptorId]
,[EntryGradeLevelReasonTypeId]
,[EntryTypeDescriptorId]
,[RepeatGradeIndicator]
,[SchoolChoiceTransfer]
,[ExitWithdrawDate]
,[ExitWithdrawTypeDescriptorId]
,[ResidencyStatusDescriptorId]
,[PrimarySchool]
,[EmployeeHiredEnrolled]
,[ClassOfSchoolYear]
,[EducationOrganizationId]
,[GraduationPlanTypeDescriptorId]
,[GraduationSchoolYear]
,[Id]
,[LastModifiedDate]
,[CreateDate]
FROM [dbo].[EdfiStudentSchoolAssociation]

```

The results table shows the following data:

StudentUSI	SchoolId	SchoolYear	EntryDate	EntryGradeLevelDescriptorId	EntryGradeLevelReasonTypeId	EntryTypeDescriptorId	RepeatGradeIndicator	SchoolCh
1		2020	2019-08-01	493	NULL	462	0	NULL
2		2020	2019-08-01	494	NULL	462	0	NULL
3		2020	2019-08-01	483	NULL	462	0	NULL
4		2020	2019-08-01	483	NULL	462	0	NULL
5		2020	2019-08-01	496	NULL	462	0	NULL
6		2020	2019-08-01	481	NULL	462	0	NULL
7		2020	2019-08-01	495	NULL	462	0	NULL
8		2020	2019-08-01	488	NULL	459	0	NULL
9		2020	2019-08-01	493	NULL	462	0	NULL
10		2020	2019-08-01	497	NULL	462	0	NULL
11		2020	2019-08-01	481	NULL	462	0	NULL
12		2020	2019-08-01	483	NULL	462	0	NULL
13		2020	2019-08-01	489	NULL	462	0	NULL

The Object Explorer on the left shows the table `dbo.EdfiStudentSchoolAssociation` highlighted with a red box and labeled "Table:". A red callout box points to the `EntryGradeLevelReasonTypeId` column in the results, asking "Why is this column NULL?".

Here is a specific example of how doing your research can help you understand why something is not working as expected and what a possible solution may be. Our member LEAs wanted to see the memo (narrative) data accompanying behavior incidents, but this data point was not being sent by their SIS vendor. We requested this data point to be sent, and the vendor agreed, but upon the release of this code we saw that some students had empty records and that there were also API errors. If we had done more research in advance, we would have known that the ODS/API only accepts up to 1,024 characters in this field, but the source system allows users to enter more than 1,024 characters. Through discussion with our member districts and with the vendor, we decided that receiving the first 1,024 characters of a memo would satisfy the end users' need, and the vendor made this adjustment in the format of how they send the data. This reduced the discovery phase of the development timeline for the vendor, because we identified the problem ourselves (although ideally we would have identified this before the initial ask).

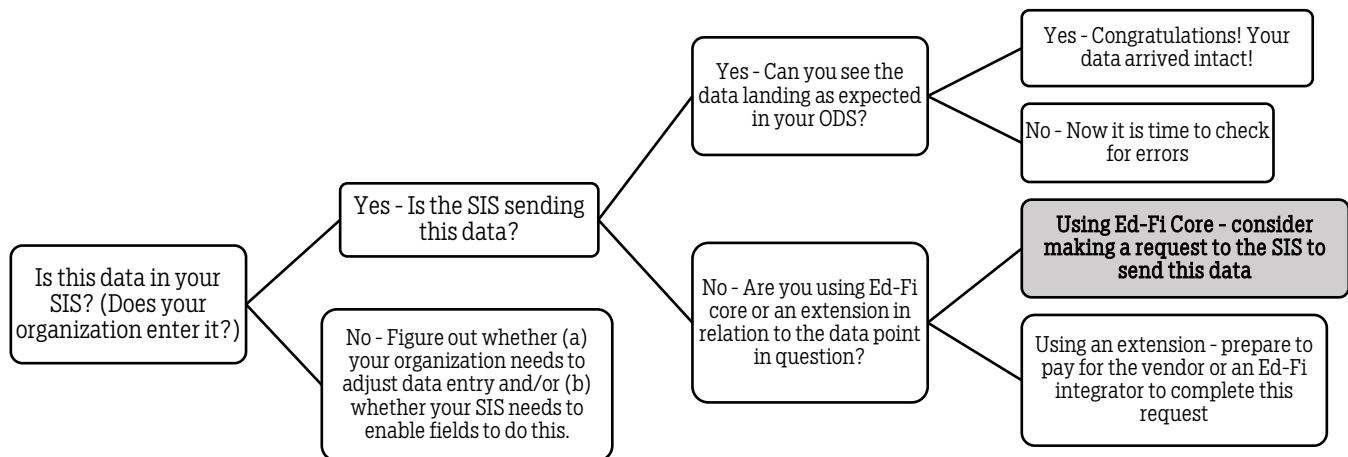
In the case that you are not sure where your data should land in the ODS, the Ed-Fi team and larger community are good resources for checking your own understanding of the documents and, if you realize that your implementation and use cases really require changes to your ODS structure, they can help you learn and apply best practices here. If you own your ODS, you do not have to adhere to anyone else's structure. You do want to keep certain design issues in mind, however; you may want to access a data element using a common ID or a table join using the Ed-Fi schema. For example, we have two tables with data related to performance levels, but there is nothing to identify the relationship between them. For us, it is not problematic because they are two different tables that have two different purposes, even though they have some common data points.

However, this could create confusion for some implementations. If you are not sure whether a particular design is best for your particular implementation, get feedback from the Ed-Fi Community and then make an informed decision.

Framing your request

If you start a conversation with your SIS vendor with “just give us X information,” and you do not know where you want that information to go in the ODS, there is a good chance that the vendor will not know where to send it. If the vendor does not know where to send the information, they could end up sending it to a location that you are not expecting, such as a related table (but not the exact table you wanted it to be in). There is no general resource on what is being sent and to where, unless the SIS vendor creates that, and some have done this.⁶ For example, PowerSchool has a wiki that has all data elements mapped and sending and the rules it considers to send all of those elements. This kind of resource is very helpful for technical staff at a district, and you can easily recognize the tables that the API is pulling from on the SIS backend and can trace it into the user interface to see why things are missing in the ODS. The SIS may have mapped it to wrong user interface (UI) element, they may not be using the UI element correctly, or they may be using a customization for the SIS that circumvents the stock UI element. Depending on the SIS product and the extent of customization options that they give to the district, the district may not even be using the standard field that the SIS is using to pull and send that data.

Here are a few concrete steps to take (see diagram below). You first need to make sure that the data you want is in the SIS (if you do not know, your SIS contact should be able to tell you). If it is in the SIS, then you need to make sure that there is a place for it to land in Ed-Fi core or that you have created an extension. Even if you do have it in the SIS, the SIS may not have created a mapping of it for Ed-Fi core, and if you are using an extension, it is unlikely that they would already have a mapping to send it there. If you just ask for it without figuring out whether it has a place to land, you are not accounting for the variables that the vendor needs to know in order to meet that request (or whether it is even possible). The SIS vendor will not know if you can receive data unless you tell them (if it is in core or an extension you have created)—do not expect them to have this information for you. Finally, if you are using an extension, you should expect to pay for this request as a customization, either to the vendor directly or using an Ed-Fi integrator. The following diagram shows you at what point you should consider making a request from the SIS.



⁶ The alternative is to look in the API logs, but that would be incredibly burdensome.

Along with this, it is essential that you can self-prioritize what requests are most important/timely and which can wait. This shows the vendor that you understand the constraints of their working environment and that you respect the time that they are devoting to your requests.

Here's a good example: You've identified that your district needs grade data to be sent as a percentage as well as a letter grade, so that schools can understand the range of student performance within the "F" grade range (0-59%) and identify students in that range for appropriate interventions. That information is being recorded in the source system, and you have also identified a specific table in the ODS that can house that data. You then submit a request to the SIS vendor highlighting the district's business need (as the client of the SIS), you include the location in the SIS where the data resides, and you send the API resource endpoint that the vendor can post that percentage grade data to. This shows that you have considered not only the relative significance of the business need for the client (that is, you have weighed which needs are worth making requests for), but you have also accounted for the SIS vendor's operating parameters as well as tried to minimize the development time necessary to meet this request on the part of the SIS vendor.

If the SIS vendor declines, it would be appropriate to ask the vendor for their reasoning as well as whether this is a request that could be met at some identifiable point in the future. A good relationship with a vendor involves them feeling comfortable communicating those kinds of parameters.

Find the right person

When you communicate with an SIS vendor, you need to be talking with the person that can help you solve your problem. Part of the job of the implementing organization is understanding the places where the source system is expecting you to record the data, so that the API sends it to the expected place in the ODS. If the implementing organization is a consortium (as we are), whose representatives are not clients or users of a specific SIS, then we will need to communicate with someone at the SIS vendor who can talk us through the details of how data and reporting is structured in the specific SIS. In our case, the person who understands that the best is usually the person responsible for state compliance. The elements recorded in the SIS are often state-reported elements. The state reporting/compliance person understands the locations and the reasons for where you want to put information in the SIS and why. This person can then interpret what we are asking for from an Ed-Fi perspective in light of potential consequences based on that change or addition. So we have asked the question in the past, if we choose to make this decision, will it influence anything in place with state reporting? (This question is only applicable if your state does not do reporting through Ed-Fi.) Having that conversation also allows the SIS vendor to adequately judge whether they will need to add a new field or new feature in order to feed this data to the ODS.

Here's another good example: Currently, our state department of education is going to request arrest data through the API. Currently neither SIS vendor has a place to record that granularity of data to send to the state. Therefore, the SIS has to either add a new field, feature, or not do it at all. These decisions will influence our decisions with ODS. For example, if the SIS vendor says they are not going to add a field and give you this ability to record arrest data in the SIS, then we as the ODS owners have to decide how to get that data from another source. Arrest data is often coming from the local sheriff's office in a spreadsheet, so our solution might be a custom data load or exploration into a direct connection with the law enforcement database.

Preferred Method of Communication

In addition to finding the right person to talk to about Ed-Fi API requests, vendors will each have their own preferred method of communication and engagement. Some may have a state-level representative or a product-level representative assigned to work with you. Identifying the preferred method of communication is important, and you should be mindful and respectful of this preference. Preferred communication may come in the form of submitting tickets, direct emails, or regular check-in calls, among other options. In addition, depending on the specific question that you have for the vendor, different representatives may need to be involved.

Make the value proposition

We have found that vendors are more receptive to requests when we can make the argument that we can make their lives easier because we are (a) increasing their client satisfaction and/or (b) able to provide some data points back to them—a common example is assessment data. The SIS could reliably pull the one or more elements they might want into the SIS, saving internal or clients' time and complexity of how to get that data element into the SIS (and reducing the possibility of human error when doing manual or upload entry). This should be a conversation involving district staff as well. These types of conversations involving technology, state reporting, and curriculum and instruction staff not only results in increased efficiency and effective data management but can also strengthen the relationships between the vendors and their district clients.

Meet them halfway

It may be tempting to believe that SIS vendors should be ready and willing to meet every request, but this expectation is not realistic. Doing your research as outlined above, prior to making requests, can help you identify the cost savings, value, and/or overall benefit of the SIS working with you. For example: If you want to make a request regarding the order of semesters, first understand value of that data, ask if they already have it located somewhere, and then if not, you can request that data because it fulfills a specific business purpose for your district or clients. If you cannot explain the business purpose and value, the SIS has more of a reason to charge you for this change; you are not making their jobs easier, and in fact you may be causing them to incur costs as opposed to showing them how working with you reduces their costs in the long run. Most cost savings related to Ed-Fi implementations are not short-term but rather are long-term, and SIS vendors may not realize all of the future cost savings at play.

On the other hand, if an SIS wants to charge you for a requested change, you need to make sure the charge is stemming from actual work. If the data is already there and they just need to add a new mapping, which is a tangible amount of work, you need to have an understanding of how much that actual work should cost and make sure that's what they are charging you (that is, make sure that the charged cost matches actual work). This means that you (the requestor) need to know how much these changes actually cost, which again means knowing your ODS and the SIS product(s). You may also need to ask other members of the Ed-Fi Community for advice on cost.

Assessment vendors

Vendors without an existing integration

First you need figure out what the assessment vendor provides in terms of exports (.csv), and then check to see if there a logical entry point in the ODS for those data points. *Logical entry point* refers to results such as raw score, scale score, proficiency levels. Then you need to figure out what data elements you need to have collocated in the ODS specifically; in other words, the purpose here is not to duplicate the work of the assessment platform, such as detailed drill-downs.

Ideally, we can persuade an assessment vendor to build an API, and the work you do beforehand in terms of mapping their data elements to the ODS data elements allows them to understand what they can send through the API a lot quicker. You should also identify the API endpoints where student data from the vendor will post. You can send them instructions as well as send your swagger site, which will have all of the required data elements that will have to be sent for data to land in the ODS. You will also need to develop meta-data and descriptors; figure out what is the bare minimum that you need to develop in order to get this assessment in, and then work backwards using the API resources that are on the swagger site. The API is basically a URL that goes all the way down to a resource which has multiple data elements, and the API will decide where the data lands in the database. So an assessment vendor does not have to keep track of the table names, it just needs to send “student assessment”, and then the API routes the data to the correct places. If you can get to a point with the vendor where you show them an API endpoint resource mapped to an existing export they already give you, you have done 30 percent of the work for API development. You have identified what data to send and where it should be sent to, and the vendor now only has to complete the coding to actually send it using whatever programming language the vendor uses.

If the vendor is not interested in building out the API, then you have to create a data mapping for data import, or you can work with an integrator to create that mapping. You’ll need to make sure that the export has some sort of unique student identifier that you can match to the ODS.

Assessment vendor with existing Ed-Fi API integration

Continue to work with them to refine the data elements that are coming out of the system to meet your use cases and your business needs. You do not want everything, you just want the most valuable. Again, you need to understand the ODS and your use case(s) to make a targeted ask to the assessment vendor. Asking them to “send everything” may end up giving you performance issues as well as many elements that you do not need. (You will likely hear district staff say “we want everything”—it is your job to figure out what they actually need.) Ed-Fi should be a complement to the platform that an assessment vendor already provides, not a duplication.

Next Steps

This document has provided discussion around key steps to take in the vendor relationship journey, but we also recommend checking out the resources provided in the footnotes as well as the Ed-Fi Implementation Playbooks,⁷ as you figure out where your vendor relationships fit into your larger implementation picture.

⁷ <https://www.ed-fi.org/blog/2019/11/introducing-ed-fi-implementation-playbooks/>