Customization and Interoperability in WME

(One author)

Abstract

One of the advantages of the Web-based Mathematics Education (WME) system is flexibility. That is, freedom for teachers to customize lesson plans and lesson material. The customized components can then be shared and distributed to anyone else using WME. However, offering such flexibility demands self-sufficiency among WME site components. Reported within are the WME component structure and how they aid methods for handling customization and interoperation.

1. Introduction and Background

With collaboration from the College of Education (Kent State University, Kent, Ohio), an impromptu approximation of a full-fledged WME site was recently constructed and deployed to a regional middle school in an effort for the continuous pragmatic assessment of the WME framework. The goal was to uncover and address WME's initial deficiencies, incorporate new functionalities, and experiment with different ways for teaching mathematical concepts [1, 2, 15]. The result is a model WME site [3] that can be deployed onto any Web server through some Web-guided installation process. In addition to the model site, the feedback information gathered allows us to correctly design and implement current projects within WME:

- *MeML and Woodpecker* --- MeML, the Mathematics Education Markup Language, is an XML application for the easy generation of WME lesson pages. Woodpecker is a prototype browser plug-in to render MeML documents [6].
- *GeoSVG* --- An SVG-based [7] geometry tool, similar to *Geometer's Sketchpad* [14]. Its abundant predefined elements make interactive animations even easier to produce [4]. Its roots in SVG inherently allow for native integration into Web documents with W3C's intermixing profile for XHTML+SVG+MathML [8].
- *DMAD* --- Distributed Mathematics Assessment Database, a massive database that is integrated with all WME sites and provides content standard student assessment data (exam questions, etc) [5].

In the beginning, this ad hoc site consisted of only a few static lesson pages. Nonetheless, it was enough to expose many flaws within our design. One of these was the failure to provide school administrators and teachers with enough flexibility for customizing certain aspects within WME. Because we were working with multiple teachers and classrooms, having only a single set of lesson pages became constraining. Teachers wanted the ability to change the wording within pages, instantly block out and reorder page sections, edit manipulatives (which are hands-on activities, interactive exercises or games), and pose their own questions for students to answer. All of these alluded to a clear indication that customization is necessary within the WME framework. We propose to take this approach a step beyond, and suggest that customizations can be reused, shared and distributed, and able to be plugged-into and exported from any other WME site as depicted in Fig. 1.

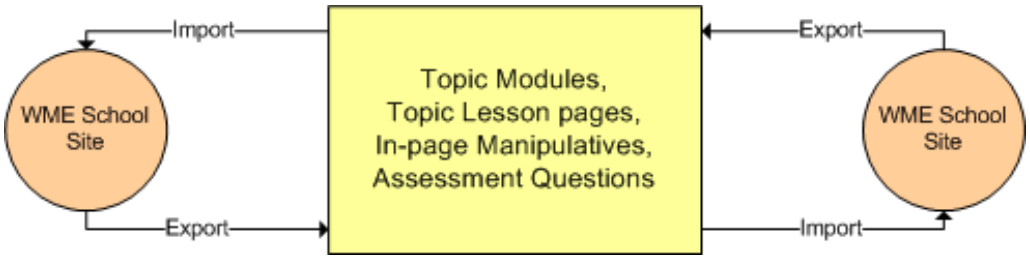

Fig. 1: An Overview of WME Site Exchange

Undoubtedly, in order to capture the above requirements, not only do naming conventions and protocol standards need to be set, the Web site itself must be built on an information architecture organized in such a way that individual[1] WME units (from Topic Modules (TM) to an in-page hands-on exercise) can be encapsulated into self-sufficient constituents. The following sections describe this approach.

2. Site Organization and Architecture for Interoperability

According to Katila and Wang, "Website information architecture (IA) deals with the structuring, the relationship, the connectivity, the logical organization, and the dynamic interactions among the constituent parts of a Web site [10]." In other words, site architecture is some blueprint, or structured map of an infrastructure for Web content placement. It is clear that a stable and standard site architecture must be maintained to handle the interoperable data encapsulation and exchange shown in Fig. 1. But before we discuss the model site architecture itself, we will first get some insight into the site's top-down organization.

Fig. 2 illustrates the per-user view of the organization of the model WME site. A user navigates from the homepage to some guided step-by-step process that begins by identifying the user's grade level, then math course under that grade level, and finally narrowing down to the user's instructor and registered course section. For a student or teacher, this interactive process leads to a dynamically generated page listing Topic Modules (TMs) and their respective Topic Lesson Pages (TLPs). Of course, these TMs are selected for use by the instructor and should probably abide by some *Content Standard* (Number Operations, Pre-algebra, Geometry, etc.) as designated by, perhaps, the NCTM [11]. The generalization of Fig. 2, however, is quite deceptive in the way that it does not show a clear relationship for the interoperable components that the WME Framework promises. Recall that WME's goals include not only the easy deployment of mathematics lessons, but also a means for handling per-user customization of these lessons, the ability to save and distribute them to another teacher or even another institution altogether. WME strives for this ease of public contribution.

[1] Here, "individual" is ambiguous: an individual Topic Module can consist of several lesson pages, an individual Topic Lesson Page is dependent on several files (ECMAScript files, graphical images, style sheets), and so on.

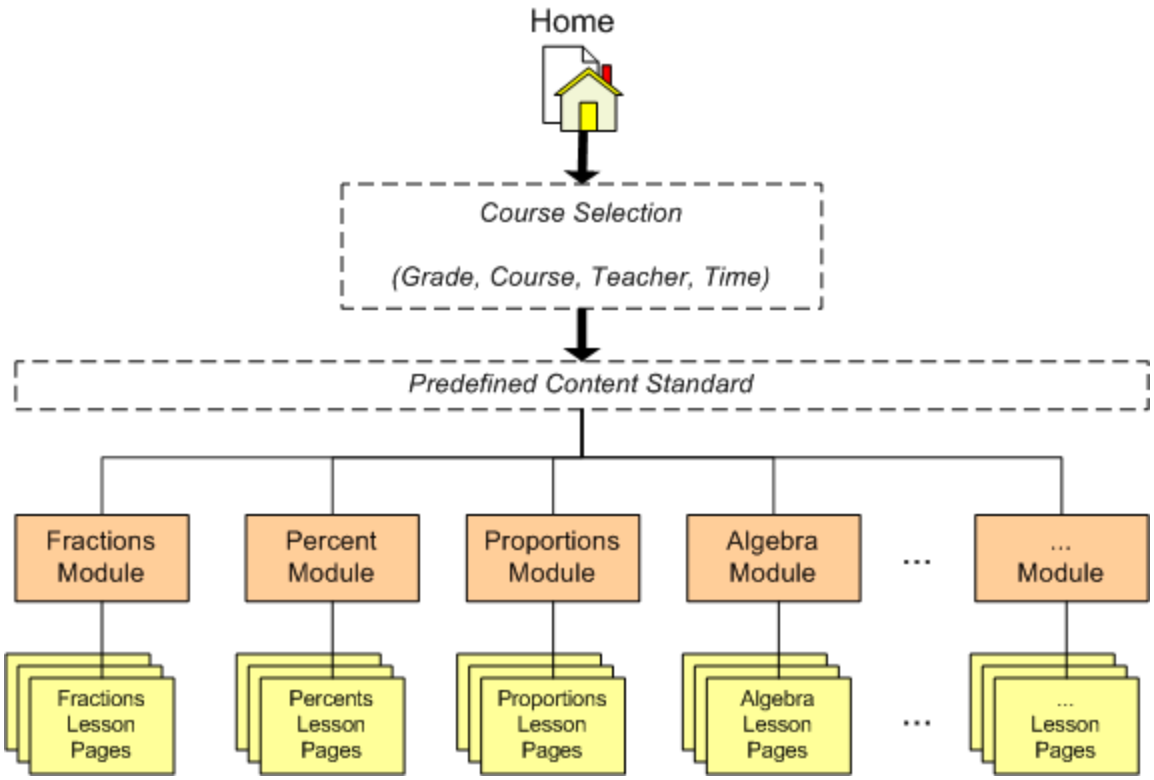

Fig. 2: A Top-down Look at the WME Model Site

This is where site architecture (Fig. 3.) comes into play. Notice the connection to a WME database. Because descriptions and characterizations such as grade levels, teachers, and their courses taught are variable across educational institutions, a persistent storage is necessary to capture and associate these values with the concrete WME components. For instance, one such relationship is the mapping of a math class to a TM with customized TLPs pertaining to the teacher's specifications. The WME site architecture and database provides a sound physical and logical organization that can allow [3]:

1. Easy means for importing and exporting TMs, TLPs, and other constituents (such as in-page sections, and manipulatives). The architecture should also be able to preserve page styles, file inclusions, graphical images before and after import/export.
2. Support for customization. As we have experienced with the pilot WME project, once a TM, TLP, etc. is imported it is unlikely that every teacher will be satisfied with the default lesson page content. Modification of these components is allowed on a per-teacher and per-class basis.

It may apparent from Fig. 3 that the architecture calls for quite an extensive file structure. That is, each TM (e.g. percent) is contained within its own directory, which then includes directories for graphical images, style sheets, and script files (not shown) for this level. Delving further into the TLP level (e.g. meal and pizza_slicing) we see that these Topic Lesson *Pages* themselves are in fact file directories rather than *pages*. Again, these directories encapsulate the content, graphical images, etc. used in a TLP (not shown). *But*

why is this file structure overkill necessary? Recall our first goal - an *easy means for importing and exporting TMs, TLPs, and other constituents.* Packaging these TMs, TLPs, and others can be done effortlessly at the directory level. Deployment and installation of these packages enjoy the same simplicity. Ubiquitous archiving tools such as *zip* can be utilized for this purpose. Another reason is that the deep directory structure causes the much desired content separation. The directory structure is necessary to capture not only default page contents, but also those customized contents. The more separation we can achieve, the more "say" and control a user has over their lessons.

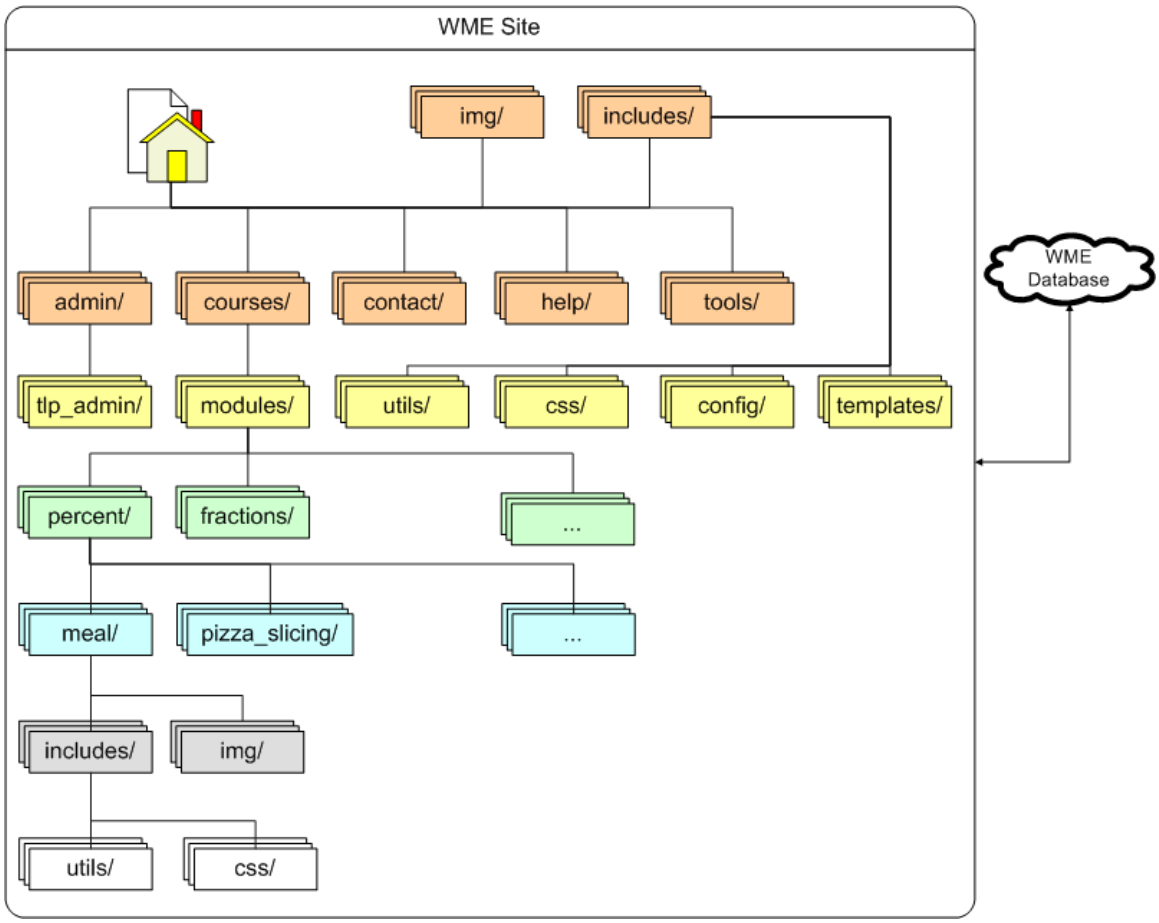

Fig. 3: The Model WME Site Architecture

3. View Sections for Content Flexibility

Save for some auxiliary Web pages for providing some institutionally related information and back-end administration, the educational substance of a WME site is essentially a composition of one or more TMs. *It is appropriate, after all, since TMs provide the lesson material!* TMs consist of TLPs packaged in the file structure as mentioned in the previous section for easy deployment. TLPs are lesson pages that convey some mathematical concept. Like TMs, TLPs are themselves product of arrangement of even smaller entities, known as *View Sections* or *VSecs*. Each VSec contains the actual page content (page markup), and can be appended with question sets (again associated per

teacher, per class). Below, Fig. 4 shows one teacher's version of a TLP conveying a lesson in statistics. Notice the separation of VSecs for content flexibility, which allows our teachers to:

- Display and hide VSecs to students at any given time for page focus.
- Instantly modify and save VSecs' wording, graphics, and manipulatives.
- Rearrange VSecs' ordering within the TLP.
- Add or delete VSecs from TLPs at will.
- Share and reuse VSecs that may or may not have been customized by teachers.

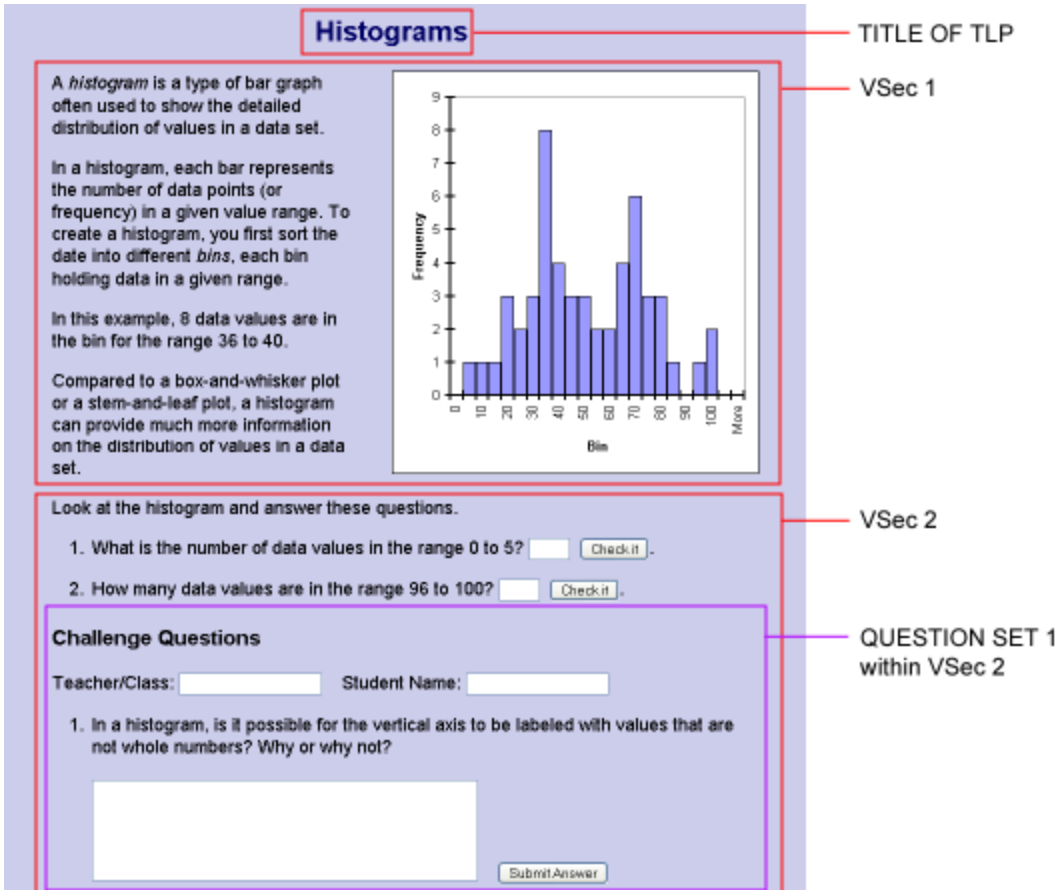

Fig. 4: TLP with Inner Elements

Because VSec manipulation happens per teacher, their "factory" versions will always remain unchanged. In fact, whenever a TM is requested by a teacher, copies of VSecs are created from default versions. Therefore, any content overwriting is avoided, and if mistakes were made, it is easy to replace VSecs with its original content. The next section discusses VSec customization.

4. Customization and Instructor Control of TLP Components

As teachers and classes are added to the WME site, opportunities for customization become available. Although teachers are welcome to use the content as provided by the

TM distributor, elements such as choice of wording, graphical images, and assessment questions may be personalized through the back-end administration area provided by the model WME site. The administrative section supplies dynamic menus tailored for three types of users: *student, teacher,* and *administrator*.

The *student menu* is quite simple because they have limited access to WME sites. Its main function consists of a course schedule that takes them directly to their course materials. Other functionalities include the set of omnipresent account configuration settings including change of passwords, names, etc.

The *teacher menu* is quite a bit more complex. A link takes them to a list of all the courses they are teaching (see Fig. 5). From the schedule, the teacher can then manage lessons or retrieve a list of enrolled students.

MATH-134 Pre-algebra

Section	Time	Location	
4	8:00am - 8:50am	111	[manage lessons] [list students]

MATH-432 Geometry

Section	Time	Location	
7	2:00pm - 2:50pm	345	[manage lessons] [list students]

MATH-222 Intermediate Math

Section	Time	Location	
8	3:00pm - 3:50pm	129	[manage lessons] [list students]

Fig. 5: A Teacher's Course Listing

If "manage lessons" is selected, the user is brought to a page where he/she can select which TMs to use for the course in question. Further penetration allows teachers to choose which TLPs to use for the course, as well as TLP customization (see Fig. 6). When changes are made, they are made immediately. This allows instructors manipulate lesson content on-the-fly that may involve any of those VSec facilities discussed in the previous section.

The *administrator menu* is geared towards school administration. These users have menus that allow them to add/delete courses, user accounts, and supply server-related technical configuration values such as database connectivity, and those involving the file system. It should be noted that user status types are stackable, that is, teachers may very receive administrative or even student statuses.

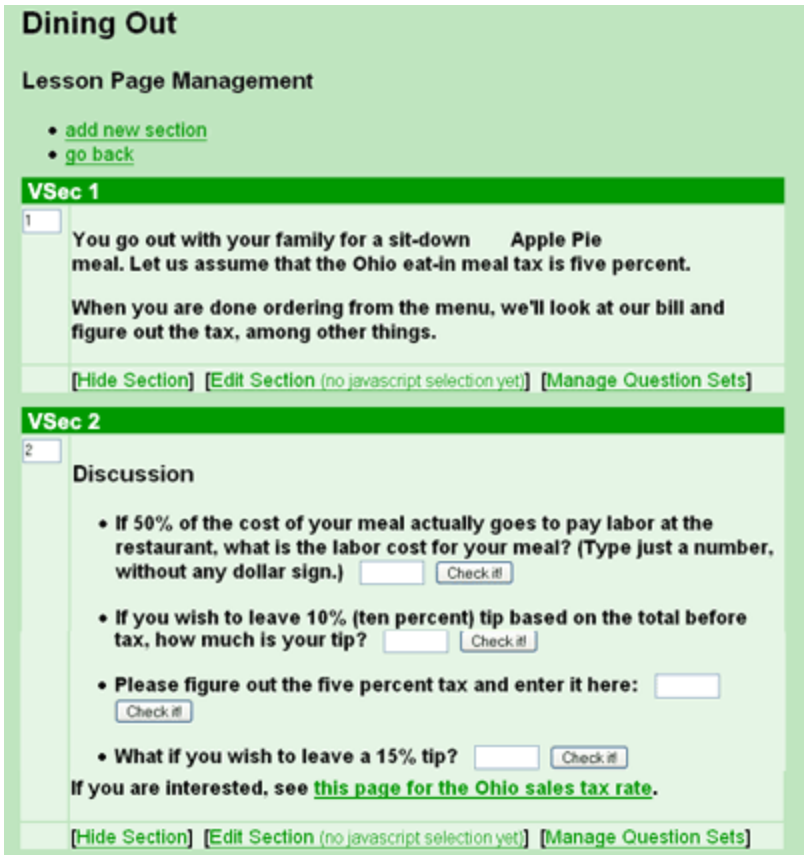

Fig. 6: TLP Customization Page

5. Conclusion and Future Work

The support for this concept of creating *and* promoting mathematics education material makes WME different from other systems - not so much in the way that we think it is *better*, but certainly more flexible. The support and facilities for lesson contribution may ambitiously lead to a Web for Mathematics Education on a global scale [12].

But to do this, WME must be fully interoperable. That is, anything designed for WME is guaranteed to operate with any other WME component. While the hierarchy and site architecture is certainly a step towards the right direction, provisions for full WME interoperability is still being investigated. One such exploration involves even deeper infiltration, past into VSec markup, seeking to allow customizable manipulatives. Other projects include:

- A *terminology finder* that links mathematical terms to definitions and TMs that supplies activities to help aid the understanding of the concept.
- *MathBoard*, a bulletin board with mathematics support (perhaps by MathML) for posing questions and responses.
- Completing implementation of *MathChat*, a live interactive forum where students can socialize with other students and teachers about mathematics in a virtual environment that simulates a physical classroom [13].

- A visual MeML editor for the intuitive creation of mathematics educational pages suitable for WME.

6. Acknowledgments

This material is based upon work supported in part by the National Science Foundation under Grant No. 0201772. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] (Five authors), "Web-based Mathematics Education Pilot Project", Proceedings of ITE'2004, McGraw-Hill, Elizabethtown, PA, September 18, 2004.

[2] (One author), "WME Site Organization and Customization Support", Proceedings of ITE'2004, McGraw-Hill, Elizabethtown, PA, September 18, 2004.

[3] (One author), "Design and Implementation of the MathChat Protocol and a Model Educational Website for the WME Framework", Master's Thesis, Department of Computer Science, Kent State University, Kent, Ohio 44242. December 2004.

[4] (Two authors), "An SVG Based Tool for Plane Geometry and Mathematics Education", submitted to IEEE SoutheastCon 2005.

[5] (Two authors), "Building DMAD: A Distributed Mathematics Assessment Database for WME", submitted to IEEE SoutheastCon 2005.

[6] (One author), "Title", submitted to IEEE SoutheastCon 2005.

[7] D. Jackson, Team Contact for the Scalable Vector Graphics Working Group, *SVG*, http://www.w3.org/TR/SVG.

[8] Masayasu, I. (ed.), "An XHTML + MathML + SVG Profile", *W3C Working Draft 9 August 2002*. Retrieved December 16, 2004 from: http://www.w3.org/TR/2002/WD-XHTMLplusMathMLplusSVG-20020809/

[10] Wang , P. S. and Katila, S.S., "An Introduction to Web Design and Programming", Brooks/Cole - Thomson Learning, Belmont, CA, 2004.

[11] National Council of Teachers of Mathematics. "Principles and Standards for School Mathematics", Reston, VA. Retrieved December 16, 2004 from: http://www.nctm.org/standards

[12] Wang, P.S., Kajler, N., Zhou, Y., Zou, X., "WME: Towards a Web for Mathematics Education", Proceedings, ISSAC'03, Aug 2003.

[13] (One author), "Web-based Mathematics Education with MathChat", Proceedings of IEEE/ITCC'2004, IEEE, Las Vegas NV, April 5-7 2004, pp. 709-717.

[14] The Geometer's Sketchpad, http://www.keypress.com/sketchpad

[15] (Six authors), "Features and Advantages of WME: a Web-based Mathematics Education System", submitted to IEEE SoutheastCon 2005.