ELECTRONIC SEISMOLOGIST

Mole: An Open Near Real-Time Database-Centric Earthworm Subsystem

by Matteo Quintiliani and Stefano Pintore

INTRODUCTION

Earthworm (Johnson *et al.*, 1995) is an automatic processing system for locating earthquakes. The initial objective of the project, started in 1993, was to provide rapid notification of seismic events. The original requirements were for a system able to manage hundreds of channels in near-real-time with robustness and speed but not necessarily to have a persistent memory of the past events. Subsequently, the Interactive Earthworm subsystem was provided in addition to the automatic, and it was made up of several modules for loading information in an Oracle database and then retrieving it for post-processing and alarm notifications (Earle *et al.*, 2003).

However, the last Earthworm public distribution containing the interactive subsystem was the 6.3 release, which dates back to May 2003, after which, for various technical and operational reasons, it was discontinued; no other database has been included in Earthworm up to the version 7.5, released in August 2011.

After several years of successful tests using Earthworm for seismic event detection in Italy (Mazza *et al.*, 2008), Istituto Nazionale di Geofisica e Vulcanologia (INGV) National Earthquake Center decided to substitute the preexisting location program with the Earthworm system, interfacing it with the current event revision procedures, the seismological database called SeisEv and the station instruments database called SeisNet. The main issue was that the common output parameters produced by Earthworm, like phase picks, event locations, magnitudes, data triggers, and error messages were reported only by e-mail or recorded on local disk files; one has to deal with scattered files in diverse formats whenever analyzing or debugging an Earthworm system.

The lack of a database in the Earthworm distribution considerably restricted the chances of analyzing data and developing new procedures for post-processing and seismic information dissemination. There already existed, and still exist, several examples of dedicated database-centric projects interfacing to Earthworm's core (Friberg *et al.*, 2010), among which the most widespread are Hydra (Buland *et al.*, 2009) and ANSS Quake Monitoring System (AQMS) (Advanced National Seismic System [ANSS], 2010; Bhadha *et al.*, 2010; Friberg and CISN developers, 2010). These projects have been developed for different purposes but they both rely on the Oracle Database Management System (DBMS). Adopting a commercial DBMS like Oracle is technically unobjectionable but this choice could be too costly for most Earthworm users, whose need could also be met with a solution based on open-source and free software currently available. Hydra, developed by USGS/NEIC (U.S. Geological Survey/National Earthquake Information Center) to locate global earthquakes and compiled only for Windows platform, uses the old Interactive Earthworm database. AQMS is not an open project at this time and even when it will be publicly released, the cost of the Oracle license should reduce the appeal of Earthworm when compared to other free earthquake monitoring systems like SeisComP3 (Olivieri and Clinton, 2012).

The experience provided by the Winston software (Cervelli *et al.*, 2004; Cervelli, 2005) had already demonstrated that a MySQL DBMS could sustain the heavy load of storing in near-real-time the waveforms coming from the Earthworm system. This fact encouraged us to rely on MySQL to develop a new system potentially capable of collecting all possible kinds of Earthworm messages.

We have designed and developed Mole, an open-source and cross-platform Earthworm subsystem made up of a MySQL database, an Earthworm module, and a web interface. Although it sprang from INGV requirements, Mole is as general-purpose as possible in order to provide a high-level communication layer to Earthworm. In this sense, Mole intends to be a starting point for the Earthworm community to fire up a development effort for a new "Open Interactive Earthworm." Mole is now freely available in the current distribution of Earthworm 7.6 released in November, 2012.

GENERAL REQUIREMENTS

The main characteristic required for our subsystem was the capability of reading the messages flowing in the Earthworm ring buffers and storing in near-real-time the contained information into a database, which had to be the expected communication layer between Earthworm and our preexisting MySQL database SeisEv.

Seismic surveillance responsibilities require that an earthquake location system must always be up and working. This could be accomplished by using redundant Earthworm instances running in parallel, which produces a new requirement: the subsystem must be capable of loading data from multiple Earthworm instances at the same time. Satisfying this

doi: 10.1785/0220120066

requirement allows centralization of information coming also from heterogeneous remote Earthworm systems set up for different purposes.

The subsystem had to be robust enough to sustain the heavy load generated during earthquake sequences. Moreover, in case of temporary database connection failures, it had to be able to buffer the loading operations and postpone their executions until the connection was reestablished.

Earthworm benefits from the constant support of a worldwide community of users and developers because it is a free system based on open-source and cross-platform software. Therefore, our subsystem and its components, such as the Database Management System, had to meet these two basic requirements so it could be readily adopted by the Earthworm community.

MOLE DEVELOPMENT

At its current state of development, Mole is able to collect in near-real-time most of the common message types flowing in the Earthworm rings and to display the stored information through a web interface. At present, Mole consists of the following the following definitions.

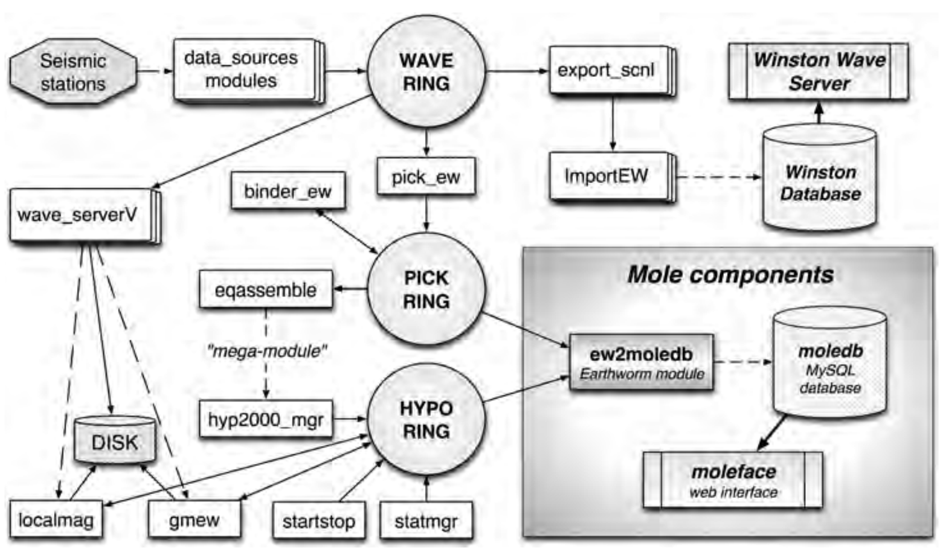
- moledb: a MySQL database which hosts Earthworm messages information. It must be created running the installation script containing all MySQL definitions.
- ew2moledb: an Earthworm module that feeds moledb with information read from the Earthworm rings. It has to be compiled for the same architecture where Earthworm is running (Solaris, Linux, Mac OS X, Windows, etc.).
- moleface: a PHP web interface for browsing the information stored in moledb. It must be installed on a web server that supports PHP and MySQL.

The very generic and minimal Earthworm diagram in Figure 1 shows Mole components and offers a view about how the single parts of the system can be connected to each other.

Because Mole must be as general-purpose as possible, we decided to keep the database schema closely related to the types of Earthworm messages, so that each message type usually matches to one or at most two corresponding tables. In this way it is easy to progressively add new message types and to implement fast-loading operations into the database. The INSERT operations are executed within MySQL stored procedures called by the Earthworm module ew2moledb. This is the only module that can write into the moledb database; therefore, other Earthworm-module developers can safely ignore database internals, and they have no need to modify their applications even if the database structure changes.

In an ideal world, in which disconnections do not occur and messages are received with a finite delay in the same order they are sent, Earthworm automatic behavior with only one ring can guarantee by itself the correct and complete sequence of the INSERT operations into the database. In reality, messages are transmitted on multiple communication channels and they could be missed, corrupted, or received in an unpredictable order. Therefore, many particular checks would be required to ensure a constant data consistency of the stored information every time data are added. Managing all possible cases in near-real-time is impracticable or even counterproductive. In practice, under normal working conditions, we can reasonably assume that the communication delays are finite (i.e., messages are received, sooner or later). So, in favor of execution speed and message-handling generality, and in order to preserve any single received piece of information, we have opted to permit stored procedures to load partial or unordered data.

If a message flows in the rings more than one time, the message will be stored in the database only the first time, and subsequently rejected as duplicated to avoid breaking referential integrity with regard to the identifiers generated by the Earthworm modules, such as pick_ew (pick_ew_XXX.ndx) and binder_ew (quake_id.d). However, nothing prevents a user from



▲ **Figure 1.** Mole components. ew2moledb can be configured to read a custom subset of message types from rings, and subsequently translate the message information and load it into the moledb database. moleface is a web interface to the moledb content.

using the Earthworm module tankplayer to play back several times the same past event, and having all messages stored in moledb.

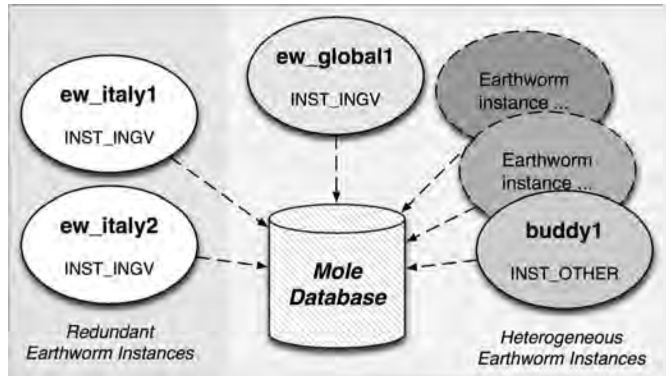
At present, Mole can handle the Earthworm message types listed in Table 1. Besides storing Earthworm messages, Mole also maps the following:

- Earthworm instances (Fig. 2), which in the context of Mole are abstractions of different Earthworm-running systems identified by unique strings—not to be mistaken for the Earthworm Installation that is usually identified by the code with prefix INST_;
- seismic SCNL codes (Station, Channel, Network, Location) and some related information such as geographic coordinates, site name, and an optional integer identifier to allow logical linking of the SCNL code to an external seismic station's database;
- names of Earthworm modules running on the different Earthworm instances, identified by "super-module-id";
- pick sequence numbers generated from different Earthworm instances, identified by "super-pick-id"; and
- earthquake location sequence numbers generated from different Earthworm instances, identified by "super-quake-id."

MySQL has been chosen from among the other open-source DBMSs because of its widespread use and to simplify management for users who already use Winston. MySQL works on all platforms on which Earthworm does; however, this is not a strict requirement as the database could be located on a remote server. Moreover, the opportunity to develop stored procedures, which MySQL natively supports starting from version 5.0, guarantees the necessary code flexibility and reusability. Stored procedure arguments in moledb are strictly determined from each Earthworm message structure. For instance, consider the following C-structure definition for the Earthworm message TYPE_PICK_SCNL:

Table 1
Current Earthworm Message Types Handled by Mole

Earthworm Message Type	Brief Description and Earthworm Module Reference
TYPE_TRACEBUF2	Produced from any kind of data_source modules, declared in Mole but not actually used because its management is delegated to the Winston Server software.
TYPE_PICK_SCNL	Pick phases usually produced by pick_ew, recently also by pick_FP.
TYPE_CODA_SCNL	Pick coda phases usually produced by the Earthworm module pick_ew.
TYPE_QUAKE2K	Event declaration and change produced by binder_ew.
TYPE_LINK	Phase/Event association produced by binder_ew.
TYPE_HYP2000ARC	Hypoinverse ARC message usually produced by hyp2000/hyp2000_mgr.
TYPE_MAGNITUDE	Local magnitude information produced by localmag.
TYPE_ STRONGMOTIONII	Ground-motion parameters produced by gmew.
TYPE_ERROR	Error status messages produced by all modules.



▲ **Figure 2.** Earthworm instances. One must define different instance identifiers (e.g.,**ew_italy1, buddy1**, etc.) for each running system, regardless of their installation identifiers (e.g., INST_INGV, INST_OTHER).

```
/* Structure to contain data from a * TYPE_PICK2K or
TYPE_PICK_SCNL. In the * comments, NTS = Null
Terminated String. */

typedef struct _EWPICK {
   /* one-byte number message type */
   unsigned char msgtype;
   /* one-byte number module id */
   unsigned char modid;
   /* one-byte number installation id */
   unsigned char instid;
   /* sequence number */
   int seq;
   /* NTS: Site code as per IRIS SEED */
   char site[TRACE2_STA_LEN];
   /* NTS: Network code as per IRIS SEED */
   char net[TRACE2_NET_LEN];
   /* NTS: Component code as per IRIS SEED */
   char comp[TRACE2_CHAN_LEN];
   /* NTS: location code as per IRIS SEED.
   * Set to "–" for TYPE_PICK2K */
   char loc[TRACE2_LOC_LEN];
   /* first-motion descriptor (U,D,' ','?')*/
   char fm;
   /* pick weight or quality (0-4) */
   char wt;
   /* time of pick - seconds since 1970 */
   double tpick;
   /* P amplitudes in digital counts */
   long pamp[3];
   } EWPICK;
```

The corresponding MySQL stored procedure sp_ins_ew_pick_scnl has arguments to accept the respective field values from the C-structure EWPICK. If ew2moledb is configured to receive TYPE_PICK_SCNL messages, it interprets the message values and calls the stored procedure like in the following example code:

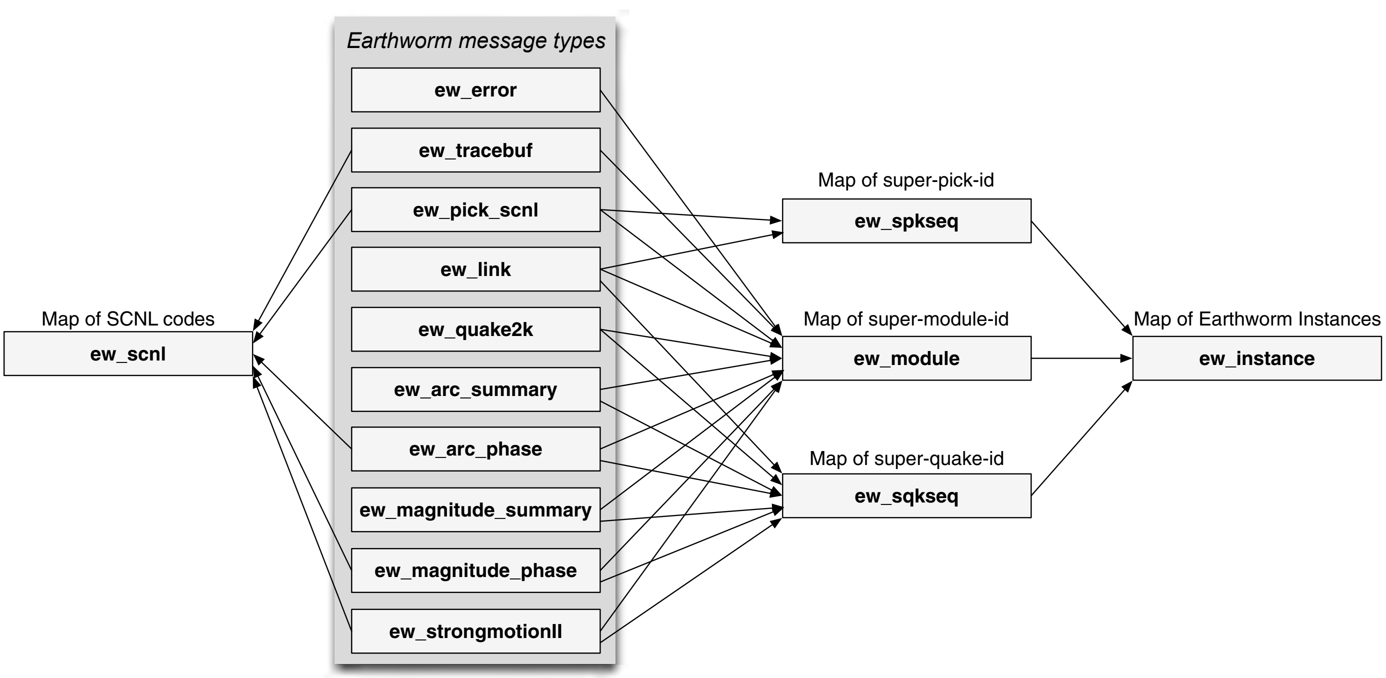
```
CALL sp_ins_ew_pick_scnl(
  'ew_italy1', 'MOD_PICK_EW', 106338,
  'IV', 'FIR', 'HHZ', '–', '?', 3,
  '2010-01-06 23:59:48.36', 2774, 1730, 1631);
```

The stored procedures are responsible for arranging information in the database tables. In the example above, the stored procedure sp_ins_ew_pick_scnl first inserts the pick sequence number of the picker module into the ew_spkseq table—whose AUTO_INCREMENT primary key is the super-pick-id—and subsequently inserts other values into the ew_pick_scnl table. If necessary, it also updates the tables ew_scnl and ew_module, containing respectively the map of SCNL station codes and module names running on different instances. It is noteworthy that ew_pick_scnl table also houses information about message type TYPE_CODA_SCNL, inserted by the stored procedure sp_ins_ew_coda_scnl. Some other Earthworm message types, such as TYPE_HYP2000ARC and TYPE_MAGNITUDE, can contain summary information followed by a list of phase or station-magnitude tuples. In moledb these kinds of messages have two corresponding tables named with the suffix _summary and _phases.

In general, the tables referring to the Earthworm message types contain, besides the various fields of the message, a few other relevant attributes such as the unique incremental id for the current table and the timestamp column modified, used for preserving the time order in which the messages have been processed. Moreover, the tables can contain other super-identifiers such as the Earthworm instance identifier ewid or the super-module-id ewmodid that have generated the message.

We have implemented the database following the same approach to generate all the other tables and following basic principles of relational database design. Figure 3 shows the current Mole database schema diagram. Adding a new message type to Mole is realized by adding the corresponding tables (one or two) in the moledb database, writing a new stored procedure to fill and update it, developing the necessary functions in ew2moledb to parse the message, and calling the corresponding stored procedure.

The internal data representation should also be designed for making straightforward queries on the entire stored information. Usually, a DBMS can provide several mechanisms for improving query performance, for instance, by defining materialized views. Although MySQL does not currently support materialized views, one may simulate them efficiently by creating and populating specific "query-oriented" tables that contain the results of predefined custom queries. Since all the messages coming from an Earthworm instance are no longer updated after their insertion, we opted for creating the "query-oriented" tables. Specific MySQL triggers update them, after the INSERT operations have been executed on the original tables of interest. An example is the table ew_events_summary that provides all-in-one earthquake locations and magnitude summary values.



▲ **Figure 3.** Mole database schema diagram. The figure displays the tables and how they are related to each other, with the boxes representing tables and the arrows foreign key relationships.

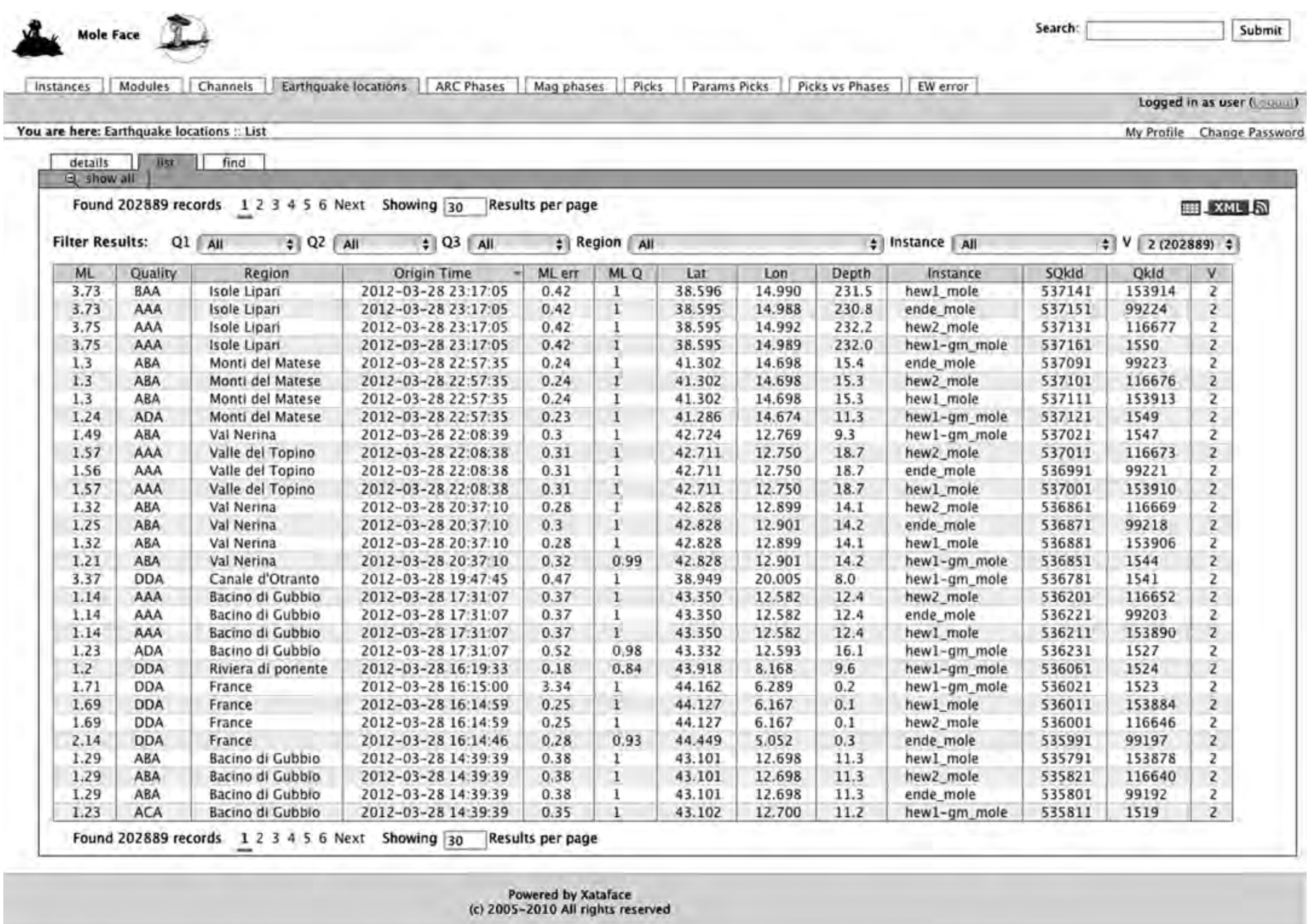
The ew2moledb module is the only additional software to be configured in every Earthworm installation in order to use Mole. No other module needs code changes, re-compilation, or configuration changes to put its messages in moledb. The ew2-moledb module needs a configuration file in which are listed the rings and the message types to read, as well as the necessary parameters for the connection to the database. An internal buffer where messages are stored temporarily until the network connection or the database becomes available again handles network disconnections to the database. This mechanism can be useful even when the database server is put down for planned maintenance.

The information contained within moledb can be displayed by the web interface called moleface, based on the open-source PHP framework Xataface (Hannah, 2009). In general, moleface allows you to browse any table, and, in particular, the homepage reports the list of the recent earthquakes, stored in the table ew_events_summary shown in the screenshot in Figure 4. From this page, one may easily apply filters based on all available fields such as magnitude, quality, and coordinates and search for whatever type of match (range, exact, partial, or multi-field). Moreover, moleface allows one to export information from moledb in the CSV (comma-separated values) and XML format or subscribe to the RSS feeds of the current search result. You can reach the extended report page for a specific location by clicking on the corresponding row in the summary list. The page displays all the details and a map of the epicenter and the stations that have contributed to the earthquake location and the magnitude estimate (Fig. 5).

IMPLEMENTATION AT INGV

Mole is currently part of Advanced Information and Data Acquisition (AIDA), the system for seismic data acquisition, analysis, archiving and distribution of the INGV National Earthquake Center.

The manual revision, web reports, e-mail, and SMS alarm procedures are based on the preexistent database SeisEv, developed at INGV and conceived for receiving any kind of information pertaining to seismic events, such as location parameters, magnitude, ShakeMaps, moment tensor and tsunamigenic characterization, but also velocity models or adminis-



ML	Quality	Region	Origin Time	ML err	ML Q	Lat	Lon	Depth	Instance	SQkId	QkId	V
3.73	BAA	Isole Lipari	2012-03-28 23:17:05	0.42	1	38.596	14.990	231.5	hew1_mole	537141	153914	2
3.73	AAA	Isole Lipari	2012-03-28 23:17:05	0.42	1	38.595	14.988	230.8	ende_mole	537151	99224	2
3.75	AAA	Isole Lipari	2012-03-28 23:17:05	0.42	1	38.595	14.992	232.2	hew2_mole	537131	116677	2
3.75	AAA	Isole Lipari	2012-03-28 23:17:05	0.42	1	38.595	14.989	232.0	hew1-gm_mole	537161	1550	2
1.3	ABA	Monti del Matese	2012-03-28 22:57:35	0.24	1	41.302	14.698	15.4	ende_mole	537091	99223	2
1.3	ABA	Monti del Matese	2012-03-28 22:57:35	0.24	1	41.302	14.698	15.3	hew2_mole	537101	116676	2
1.3	ABA	Monti del Matese	2012-03-28 22:57:35	0.24	1	41.302	14.698	15.3	hew1_mole	537111	153913	2
1.24	ADA	Monti del Matese	2012-03-28 22:57:35	0.23	1	41.286	14.674	11.3	hew1-gm_mole	537121	1549	2
1.49	ABA	Val Nerina	2012-03-28 22:08:39	0.3	1	42.724	12.769	9.3	hew1-gm_mole	537021	1547	2
1.57	AAA	Valle del Topino	2012-03-28 22:08:38	0.31	1	42.711	12.750	18.7	hew2_mole	537011	116673	2
1.56	AAA	Valle del Topino	2012-03-28 22:08:38	0.31	1	42.711	12.750	18.7	ende_mole	536991	99221	2
1.57	AAA	Valle del Topino	2012-03-28 22:08:38	0.31	1	42.711	12.750	18.7	hew1_mole	537001	153910	2
1.32	ABA	Val Nerina	2012-03-28 20:37:10	0.28	1	42.828	12.899	14.1	hew2_mole	536861	116669	2
1.25	ABA	Val Nerina	2012-03-28 20:37:10	0.3	1	42.828	12.901	14.2	ende_mole	536871	99218	2
1.32	ABA	Val Nerina	2012-03-28 20:37:10	0.28	1	42.828	12.899	14.1	hew1_mole	536881	153906	2
1.21	ABA	Val Nerina	2012-03-28 20:37:10	0.32	0.99	42.828	12.901	14.2	hew1-gm_mole	536851	1544	2
3.37	DDA	Canale d'Otranto	2012-03-28 19:47:45	0.47	1	38.949	20.005	8.0	hew1-gm_mole	536781	1541	2
1.14	AAA	Bacino di Gubbio	2012-03-28 17:31:07	0.37	1	43.350	12.582	12.4	hew2_mole	536201	116652	2
1.14	AAA	Bacino di Gubbio	2012-03-28 17:31:07	0.37	1	43.350	12.582	12.4	ende_mole	536221	99203	2
1.14	AAA	Bacino di Gubbio	2012-03-28 17:31:07	0.37	1	43.350	12.582	12.4	hew1_mole	536211	153890	2
1.23	ADA	Bacino di Gubbio	2012-03-28 17:31:07	0.52	0.98	43.332	12.593	16.1	hew1-gm_mole	536231	1527	2
1.2	DDA	Riviera di ponente	2012-03-28 16:19:33	0.18	0.84	43.918	8.168	9.6	hew1-gm_mole	536061	1524	2
1.71	DDA	France	2012-03-28 16:15:00	3.34	1	44.162	6.289	0.2	hew1-gm_mole	536021	1523	2
1.69	DDA	France	2012-03-28 16:14:59	0.25	1	44.127	6.167	0.1	hew1_mole	536011	153884	2
1.69	DDA	France	2012-03-28 16:14:59	0.25	1	44.127	6.167	0.1	hew2_mole	536001	116646	2
2.14	DDA	France	2012-03-28 16:14:46	0.28	0.93	44.449	5.052	0.3	ende_mole	535991	99197	2
1.29	ABA	Bacino di Gubbio	2012-03-28 14:39:39	0.38	1	43.101	12.698	11.3	hew1_mole	535791	153878	2
1.29	ABA	Bacino di Gubbio	2012-03-28 14:39:39	0.38	1	43.101	12.698	11.3	hew2_mole	535821	116640	2
1.29	ABA	Bacino di Gubbio	2012-03-28 14:39:39	0.38	1	43.101	12.698	11.3	ende_mole	535801	99192	2
1.23	ACA	Bacino di Gubbio	2012-03-28 14:39:39	0.35	1	43.102	12.700	11.2	hew1-gm_mole	535811	1519	2

Found 202889 records 1 2 3 4 5 6 Next Showing 30 Results per page

Powered by Xataface
(c) 2005–2010 All rights reserved

▲ **Figure 4.** Moleface earthquake locations summary. Moleface homepage screenshot shows a summary list of recent earthquake locations produced by the different Earthworm instances. One may filter the list choosing the available values in the combo boxes at the top of table. One may also make more complex queries with the Find Form, searching range or exact values.

▲ **Figure 5.** Moleface earthquake location details. The page reports all available information about a single earthquake location and shows on the map the epicenter and the stations contributing only to location (triangle), only to magnitude estimate (square with flag), or both (circle). You can reach detailed information about the location and magnitude phases by clicking on the corresponding link.

trative areas. The schema of the database moledb is implicitly governed by Earthworm messages, thus containing only a subset of all the information stored in SeisEv. Accordingly, we use Mole as the communication layer for transporting information from Earthworm to the main INGV system. This task has been accomplished hosting both databases on the same MySQL server and developing customized triggers on moledb to feed SeisEv tables. On the same server also resides the database SeisNet where all the information regarding seismic stations is stored. An adhoc procedure retrieves this information and creates the Earthworm configuration files, updating the ew_scnl table of moledb with the current values of geographic coordinates, site name, and primary key of the channel table in SeisNet.

Our MySQL server is based on two physical machines installed at INGV headquarters in Rome. It is implemented as a high-availability solution using DRBD for file-system replication (http://www.drbd.org/) and HeartBeat for automatic failover support (http://www.linux-ha.org). A twin system is located at our Disaster Recovery Center in Grottaminarda (about 280 km southeast of Rome). The database server at Grottaminarda is currently switched off, but as soon as the ongoing work to improve the connectivity between the two data centers is completed, the twin MySQL servers will eventually be set in a master–master configuration to obtain database replication. MySQL is capable of scaling to a multi-master architecture, and this is also a planned development for AIDA.

We currently have four homogenous Earthworm instances running in parallel and writing into the moledb database. The procedures in AIDA are able to revise seismic locations produced by any Earthworm instance, but for convenience we always use one of them during scheduled maintenance downtime or as preferred instance unless a major failure occurs.

AIDA is 24/7 fully operational from 4 May 2012, collecting and analyzing data in near-real-time from more than 400 stations, totaling about 1,700 channels. On 20 May 2012 a M_L 5.9 earthquake occurred in northern Italy and started a sequence that produced more than 2,100 events over 40 days (Mazza *et al.*, 2012). During this time, any single Earthworm instance has successfully inserted in moledb about 12,000 automatic locations of earthquakes in Italy, with magnitude and strong motion messages.

SUMMARY AND FUTURE DEVELOPMENTS

We have developed Mole, an open-source and cross-platform Earthworm subsystem able to store Earthworm messages in a database. It allows higher service continuity by dealing with multiple Earthworm instances running in parallel. Mole relies on the MySQL DBMS, and the ew2moledb module can use its own buffering system to cope with long-term disconnections from the DBMS server caused either by a server issue or by planned database maintenance. The moleface web interface is provided for easy inspection of moledb database contents.

At INGV National Earthquake Center, Mole has allowed us to substitute the old home-made automatic earthquake-location system with Earthworm, leaving nearly unchanged the preexisting service framework for the Italian Seismic Surveillance. Additionally, it has made the achievement of service continuity, data replication, and disaster recovery easier.

The moledb database is used to feed the preexisting INGV SeisEv database through MySQL's built-in trigger mechanism. A similar approach can be adopted to move Earthworm information in near-real-time to any other seismological MySQL database. In future, moledb could be extended through community efforts toward a more complete database, including other seismological information (e.g., complete station information or Earthworm configuration parameters), and by developing new procedures, real-time or offline, to extract and transfer information using standard formats for data exchange, such as QuakeML (Schorlemmer *et al.*, 2011).

Mole is publicly distributed as a free software and its sources are available from the official Earthworm repository (http://earthworm.isti.com/trac/earthworm/). The open and free database interfacing it can boost the community contributions toward a new Open Interactive Earthworm for near real-time monitoring of Earthworm operations and manual revision of earthquake parameters.

ACKNOWLEDGMENTS

We thank Salvatore Mazza and those we had the pleasure of sharing the work on AIDA: A. Bono, V. Lauciani, A. Mandiello, C. Marcocci, F. Mele and L. Scognamiglio. We are grateful to Paul Friberg and all the people at ISTI (Instrumental Software Technologies, Inc.) for their indispensable feedbacks and ceaseless support to the Earthworm user and developer community. We also thank the anonymous reviewer for the careful reading of our manuscript and thoughtful comments.

REFERENCES

Advanced National Seismic System (ANSS) (2010). AQMS: Advanced National Seismic System Quake Management System, http://vault.gps.caltech.edu/trac/cisn.

Bhadha, R., S. Chen, J. Crummey, E. Hauksson, K. V. I. Thomas, M. Watkins, R. Yip, E. Yu, D. Given, R. Peats, and S. Schwarz (2010). Caltech/USGS southern California seismic network: Recent developments (abstract S13C-2018), *AGU* (Fall Meet.), S13C-2018.

Buland, R. P., M. Guy, D. Kragness, J. Patton, B. Erickson, M. Morrison, C. Bryon, D. Ketchum, and H. Benz (2009). Comprehensive seismic monitoring for emergency response and hazards assessment: Recent developments at the USGS National Earthquake Information Center (abstract S11B-1696), *AGU* (Fall Meet.), S11B-1696.

Cervelli, D. P., P. F. Cervelli, and T. L. Murray (2004). New software for long-term storage and analysis of seismic wave data (abstract SF13A-0705), *AGU* (Fall Meet.), SF13A-0705.

Cervelli, D. P. (2005). Winston Manual, http://www.avo.alaska.edu/Software/winston/W_Manual_TOC.html.

Earle, P., A. Bittenbinder, B. Bogaert, and C. Johnson (2003). Turn to the worm: Seismic network operation using the USGS Earthworm system, in *Observations and Research Facilities for European Seismology (ORFEUS) Electronic Newsletter*, Vol. 5, no. 1, http://www.orfeus-eu.org/Organization/Newsletter/vol5no1/earthworm.html.

Friberg, P., S. Lisowski, I. Dricker, and S. Hellman (2010a). Earthworm in the 21st century, in *European Geosciences Union General Assembly*, 2–7 May 2010, Vienna, Austria, 12654 pp.

Friberg, P., and CISN developers (2010). A Reorganization of earthquake post-processing systems at regional seismic networks in the United States using AQMS, in *European Seismological Commission (ESC)*, 6–10 September 2010, Abstract S21/Tu/O13.

Hannah, S. (2009). Xataface, a better way to build a front-end for your MySQL database, http://www.xataface.com.

Johnson, C. E., A. Bittenbinder, B. Bogaert, L. Dietz, and W. Kohler (1995). Earthworm: A flexible approach to seismic network processing, *Incorporated Research Institutions for Seismology (IRIS) Newsletter* **14,** 1–4.

Mazza, S., A. Basili, A. Bono, V. Lauciani, A. Mandiello, C. Marcocci, F. Mele, S. Pintore, M. Quintiliani, L. Scognamiglio, and G. Selvaggi (2012). AIDA—Seismic data acquisition, processing, storage and distribution at the National Earthquake Center, INGV, *Ann. Geophys.* **55,** no. 4, doi: 10.4401/ag-6145.

Mazza, S., F. Mele, M. Olivieri, S. Pintore, M. Quintiliani, L. Scognamiglio, and G. Selvaggi (2008). A new system for earthquake detection and source parameters determination in Italy, in *European Seismological Commission (ESC)*, 7–12 September 2008, Crete, Greece.

Olivieri, M., and J. Clinton (2012). An almost fair comparison between Earthworm and SeisComp 3, *Seismol. Res. Lett.* **83,** no. 4, 720–727.

Schorlemmer, D., F. Euchner, P. Kästli, and J. Saul (2011). QuakeML: Status of the XML-based seismological data exchange format, *Ann. Geophys.* **54,** no. 1, 59–65, doi: 10.4401/ag-4874.

Matteo Quintiliani
Stefano Pintore
Istituto Nazionale di Geofisica e Vulcanologia
Via di Vigna Murata
605-00143 Roma, Italy
matteo.quintiliani@ingv.it