Natural Language Interaction Protocol

IBM

Cisco

Red Hat

Hitachi

ServiceNow

Fordham University

U. Michigan

Purdue University

U. Delaware

U. Buffalo

Pennsylvania State University

Indiana University

SRI International

Collaborators

- IBM Research
  - Dinesh Verma
  - Abhay Ratnaparkhi
  - Jonathan Lenchner
  - Ranjan Sinha
- RedHat
  - Eric Erlandson
  - Sanjay Aiyagari
- Cisco
  - Ashish Kundu
- Hitachi
  - Yohei Kawaguchi

- ServiceNow
  - Sean Hughes
- SRI International
  - Yan-Ming Chiou
- Independents
  - Raj Doodhiawala
  - Tom Sheffler
- U. Delaware
  - Chien-Chung Shen
  - Mathews Mauriello
- U. at Buffalo
  - Jinjun Xiong

- U. Michigan
  - Sugih Jamin
- Purdue University
  - Elisa Bertino
- Indiana University
  - Luyi Xing
- Fordham
  - Mahomed Rahouti
- Pennsylvania State University
  - Winpeng Yin

Motivation

There will be several multi-agent systems working within an enterprise and across enterprises

These agents need a standard common protocol to communicate

NLIP provides a standard common protocol for agents to communicate with each other

NLIP

NLIP
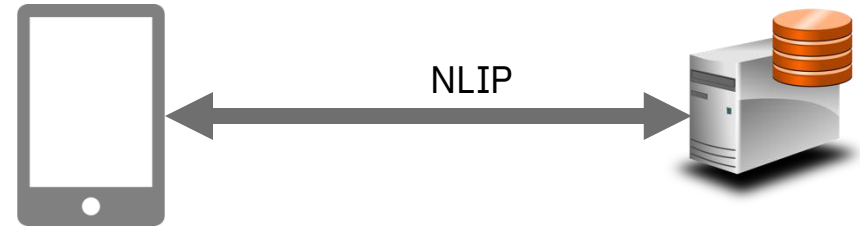
AI enables a new common protocol for communication

When we need to conduct our affairs with various types of businesses

– Banks

– Retail Shops

– Telecommunication Vendors

– Restaurants etc.

We just use one common language to interact with the business person on the other side

– And the common language is good enough to get all the work done

When computers in our control (our phones/personal devices) need to conduct our affair with various types of businesses

– Banks

– Retail Shops

– Telecommunication Vendors

– Restaurants etc.

Each business server uses its own proprietary application-level protocol to communicate with the clients

– Resulting in an explosion of mobile apps and application level protocols

History Repeats Itself

Observation

- The IT landscape today is similar to that of 1990s in some ways

- 1990s:

  - A plethora of client side applications supporting various business solutions

  - The emergence of a simple standard protocol (HTTP) replaced these with a single client application

  - Simplifies business solution delivery task, and increased Internet Commerce

Today

- A plethora of mobile applications supporting various business solutions

- The emergence of LLM leads to the use of chat as a common interface to business solutions

If we can define a single standard protocol for a single mobile application to talk to business servers

- We can simplify the delivery of business solutions

- We can increate the uptake of genAI for B2B and B2C.

NLIP will be for generative AI technology what HTTP was for the Internet technology.

The Vision

A Simple Standard Protocol between a mobile application and a server-side business service

– Enables an intuitive multimodal interaction

– Assumes genAI capability on both end-points

NLIP
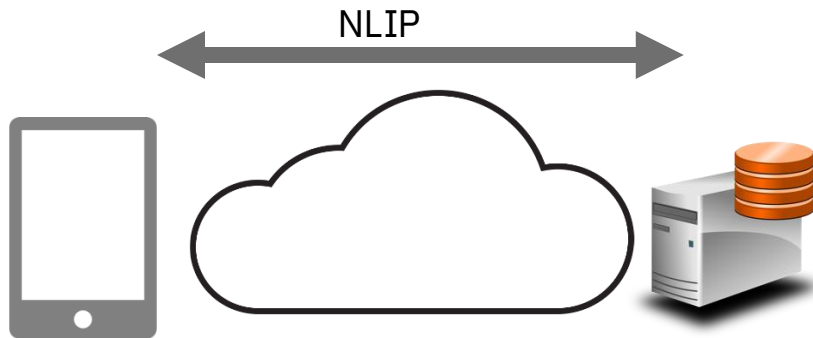
Requirements

– Protocol should be secure

– Should enable various safeguards

  • data privacy, data usage policy, DDoS prevention

– Protocol should be implementable over various underlying transports (REST/QUIC/…)

– Should support multimedia content exchange

– Should permit communication efficiency

– Should enable implementation in various languages

– Should be standardized in an open forum

NLIP: Salient Features

Promotes an intuitive human-centric communication

- Primary mode of interaction is natural language

  - Could be typed text or spoken text with speech to text conversion

- Vision, Audio and other modalities are supported

  - Some business interactions require transmission of images and videos

- Structured text supported but not intended as a primary interaction mode

- No custom configuration – All configuration and policy exchanges made using natural language

- The AI models used by different end-points (clients and servers) in a NLIP session can be different

NLIP Scope

In-Scope

- Communication between two AI-Agents

  - Both across organizations and within organization

- Communication between a human and an AI-Agent

  - Both across organizations and within organization

Out of Scope

- Communication between legacy software

  - Both across organizations and within organization

- Communication between a legacy software (not using AI) client and an AI-Agent

  - Both across organizations and within organization

Working Definition of AI Agent: An AI Agent is software that uses an AI Model (LLM or various modalities) or can call out to an AI Model to interpret general text, vision and other modalities.

The Initial Specification

NLIP defined as a layer above HTTPS/REST

– Can leverage other equivalent layers as well

Assumes the following functionality from the
underlying layer

– A request response paradigm

– Encrypted communication of information

– Support to carry authentication tokens

NLIP is developed around an exchange of JSON
messages

NLIP

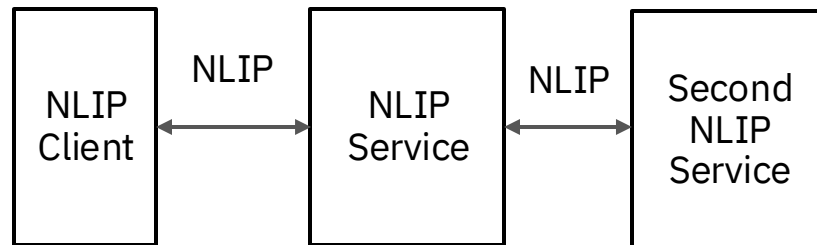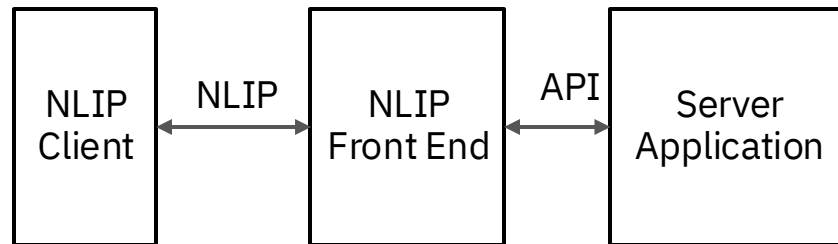Underlying Protocol
(e.g. HTTPS/REST)
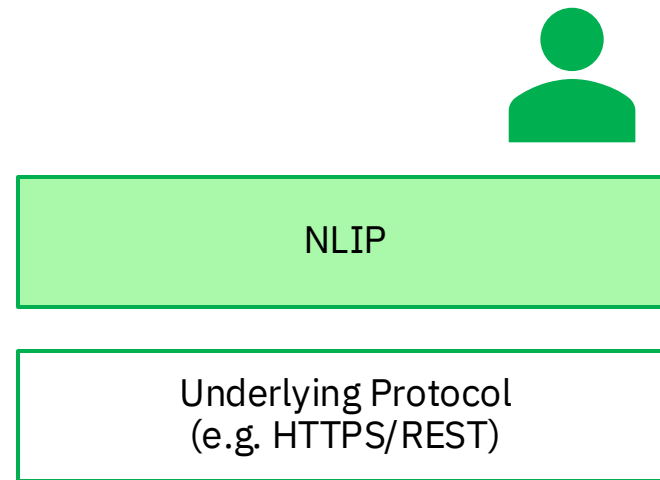
NLIP In Action

The typical interaction would be between a NLIP client and a NLIP front-end

– With the backend being typical business service

NLIP messages would trigger an interaction with the backend

– The Backend could be a traditional API based service

– The backend could be another LLM based chat-engine

– The backend could be another NLIP enabled service

```
┌──────────┐         ┌──────────┐         ┌──────────┐
│  NLIP    │  NLIP   │  NLIP    │  API    │ Server   │
│  Client  │◄──────► │ Front End│◄──────► │Application│
└──────────┘         └──────────┘         └──────────┘
```

```
┌──────────┐         ┌──────────┐         ┌──────────┐
│  NLIP    │  NLIP   │  NLIP    │  NLIP   │ Second   │
│  Client  │◄──────► │ Service  │◄──────► │  NLIP    │
└──────────┘         └──────────┘         │ Service  │
                                          └──────────┘
```
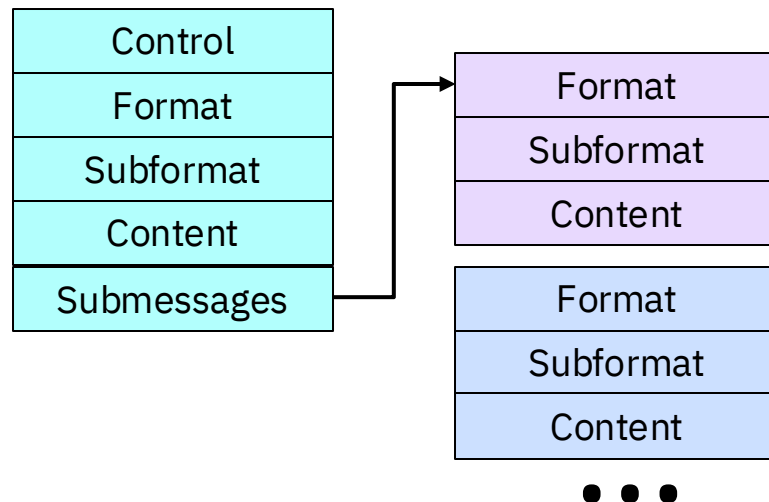
NLIP Messages

The basic structure of an NLP Exchange is a Sub-Message

- Format: whether content is text, structured content, image etc.

- Subformat: Specialization of format, e.g. language for text, encoding for images etc.

- Content: The actual content

- An optional synchronization field

NLIP Request and Response Messages consist of a sequence of sub-messages.

- And indicates if it is a control message

  • A control message could inquire about policies of the service, configuration exchange etc.

Since most messages will have only one sub-message, the NLIP message contains the first format, subformat and content at very top with an optional semantics indicator for synchronization.

Control
Format
Subformat
Content
Submessages

Format
Subformat
Content

Format
Subformat
Content

● ● ●

Current Set of NLIP Messages

Format	Subformat	Notes
text	Language of text (e.g. English, French etc.)	Provides a hint to the other side, can be used for model selection
token	An opaque string – determined by token creator	Content is an opaque string – used by end-point to share information such as correlators or authentication data
structured	JSON, URI, XML	Content is structured format. Allows backward compatibility for display of graphical GUI
binary	Encoding – image, video, sensor etc.	Subformat is content-type/encoding
location	Text or GPS	Coordinate if GPS, description if text
generic	Open	Provided for future extensions

NLIP Reference Architecture
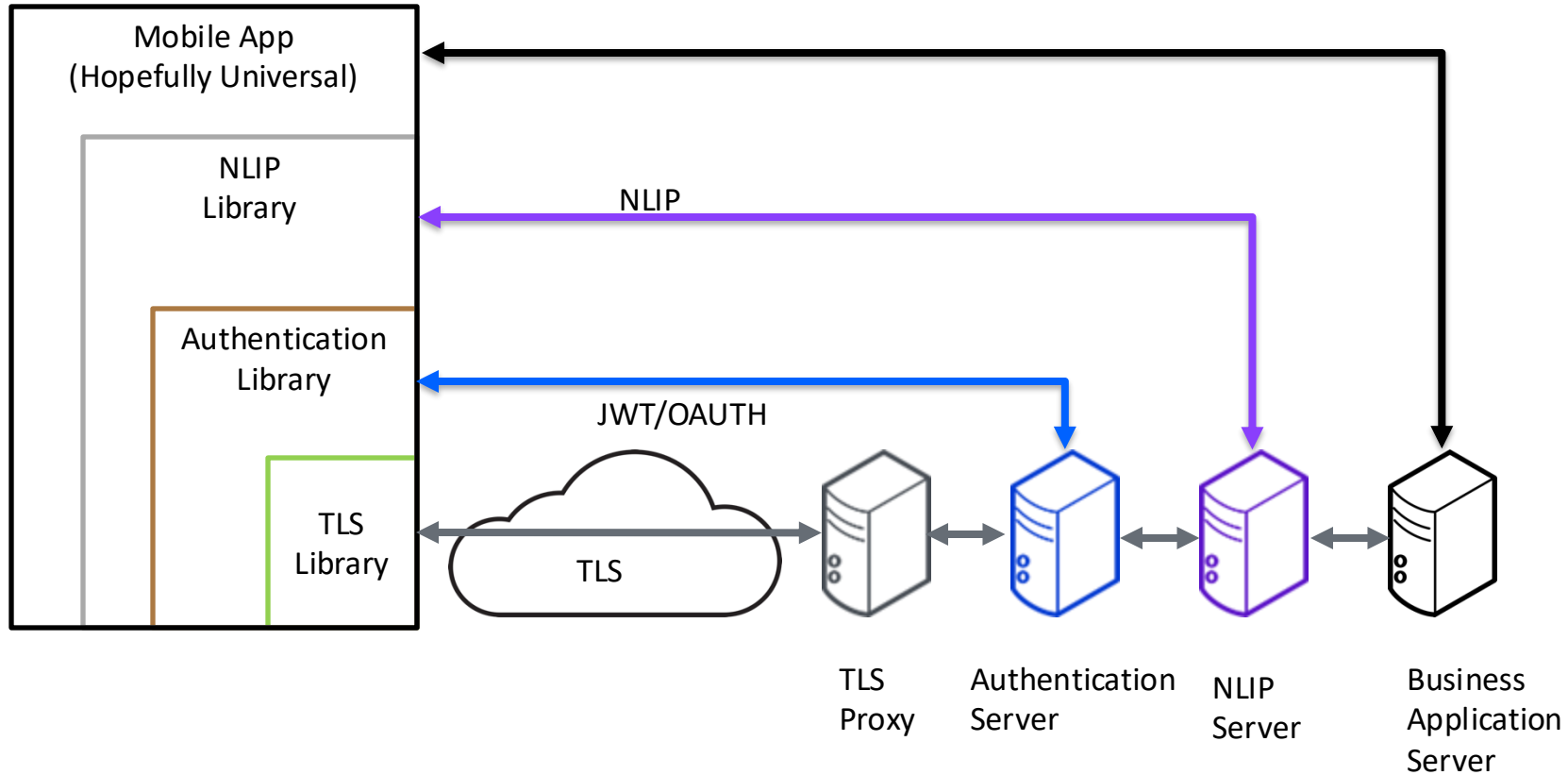
This architecture can be implemented in different strategies - merging/splitting servers and libraries

NLIP Differentiators

NLIP enables a single API to be offered by different agents for their functions

NLIP focuses on the modality of the content to be exchanged between agents

– relying on the LLM capability of the agents to interpret the content and take actions thereof

Status

NLIP being defined and standardized under ECMA TC-56

Draft Specifications and Initial implementations at https://github.com/nlip-project

Plan is to establish an official standard in 2H 2025.

THANK YOU